Nos. 2023-1850, -2038

United States Court of Appeals
for the
Fourth Circuit

HONEYWELL INTERNATIONAL INC.; HAND HELD PRODUCTS, INC.; METROLOGIC INSTRUMENTS, INC.,

Plaintiffs-Appellants/Cross-Appellees,

v.

OPTO ELECTRONICS CO., LTD.,

Defendant-Appellee/Cross-Appellant.

On Appeal from the United States District Court for the Western District of North Carolina Case No. 3:21-cv-506-KDB-DCK

JOINT APPENDIX
VOLUME 7 OF 16 (PAGES 3031-3214)

Kirk T. Bradley
M. Scott Stevens
S. Benjamin Pleune
Stephen R. Lareau
Nicholas C. Marais
Lauren N. Griffin
Brandon C.E. Springer
ALSTON & BIRD LLP
Vantage South End
1120 South Tryon Street, Ste. 300
Charlotte, NC 28203
(704) 444-1000

Attorneys for Plaintiffs-Appellants Honeywell International Inc.; Hand-Held Products, Inc.; and Metrologic Instruments, Inc.

York M. Faulkner
YORKMOODYFAULKNER
711
Tensho Building, 7F
3-17-11 Shibaura
Tokyo, Minato-Ku, 108-0023
Japan
(202) 251-0837

Brian D. Schmalzbach
MCGUIREWOODS, LLP
800 East Canal Street
Richmond, Virginia 23219
(804) 775-4746

Attorneys for Defendant-Appellee Opto Electronics Co., Ltd.

April 1, 2024

VOLUME 1 OF 16

INTERNATIONAL STANDARD

ISO/IEC 15415

Second edition
2011-12-15

Reviewed and confirmed in 2018

Information technology — Automatic identification and data capture techniques — Bar code symbol print quality test specification — Two-dimensional symbols

Technologies de l'information — Techniques automatiques d'identification et de capture des données — Spécification de test de qualité d'impression des symboles de code à barres — Symboles bidimensionnels

EXHIBIT
1004

Reference number
ISO/IEC 15415:2011(E)

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 15415 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 31, *Automatic identification and data capture techniques*.

This second edition cancels and replaces the first edition (ISO/IEC 15415:2004), which has been technically revised. It also incorporates the Technical Corrigendum ISO/IEC 15415:2004/Cor.1:2008.

Introduction

The technology of bar coding is based on the recognition of patterns encoded, in bars and spaces or in a matrix of modules of defined dimensions, according to rules defining the translation of characters into such patterns, known as the symbology specification. Symbology specifications may be categorised into those for linear symbols, on the one hand, and two-dimensional symbols on the other; the latter may in turn be sub-divided into “multi-row bar code symbols”, sometimes referred to as “stacked bar code symbols”, and “two-dimensional matrix symbols”. In addition, there is a hybrid group of symbologies known as “composite symbologies”; these symbols consist of two components carrying a single message or related data, one of which is usually a linear symbol and the other a two-dimensional symbol positioned in a defined relationship with the linear symbol.

Multi-row bar code symbols are constructed graphically as a series of rows of symbol characters, representing data and overhead components, placed in a defined vertical arrangement to form a (normally) rectangular symbol, which contains a single data message. Each symbol character has the characteristics of a linear bar code symbol character and each row has those of a linear bar code symbol; each row, therefore, may be read by linear symbol scanning techniques, but the data from all the rows in the symbol must be read before the message can be transferred to the application software.

Two-dimensional matrix symbols are normally square or rectangular arrangements of dark and light modules, the centres of which are placed at the intersections of a grid of two (sometimes more) axes; the coordinates of each module need to be known in order to determine its significance, and the symbol must therefore be analysed two-dimensionally before it can be decoded. Dot codes are a subset of matrix codes in which the individual modules do not directly touch their neighbours but are separated from them by a clear space.

Unless the context requires otherwise, the term “symbol” in this International Standard may refer to either type of symbology.

The bar code symbol must be produced in such a way as to be reliably decoded at the point of use, if it is to fulfil its basic objective as a machine-readable data carrier.

Manufacturers of bar code equipment and the producers and users of bar code symbols therefore require publicly available standard test specifications for the objective assessment of the quality of bar code symbols (a process known as verification), to which they can refer when developing equipment and application standards or determining the quality of the symbols. Such test specifications form the basis for the development of measuring equipment for process control and quality assurance purposes during symbol production as well as afterwards.

The performance of measuring equipment for the verification of symbols (verifiers) is the subject of a separate International Standard (ISO/IEC 15426, Parts 1 and 2).

This International Standard is intended to achieve comparable results to the linear bar code symbol quality standard ISO/IEC 15416, the general principles of which it has followed. It should be read in conjunction with the symbology specification applicable to the bar code symbol being tested, which provides symbology-specific detail necessary for its application. Two-dimensional multi-row bar code symbols are verified according to the ISO/IEC 15416 methodology, with the modifications described in Clause 6; different parameters and methodologies are applicable to two-dimensional matrix symbols.

There are currently many methods of assessing bar code quality at different stages of symbol production. The methodologies described in this International Standard are not intended as a replacement for any current process control methods. They provide symbol producers and their trading partners with universally standardized means for communicating about the quality of multi-row bar code and two-dimensional matrix symbols after they have been printed. The procedures described in this International Standard must necessarily be augmented by the reference decode algorithm and other measurement details within the

applicable symbology specification, and they may also be altered or overridden as appropriate by governing symbology or application specifications.

Alternative methods of quality assessment may be agreed between parties or as part of an application specification.

For direct part mark applications, a modified version of the methodology defined in this International Standard has been defined in ISO/IEC TR 29158.

Information technology — Automatic identification and data capture techniques — Bar code symbol print quality test specification — Two-dimensional symbols

1 Scope

This International Standard

— specifies two methodologies for the measurement of specific attributes of two-dimensional bar code symbols, one of these being applicable to multi-row bar code symbologies and the other to two-dimensional matrix symbologies;

— defines methods for evaluating and grading these measurements and deriving an overall assessment of symbol quality;

— gives information on possible causes of deviation from optimum grades to assist users in taking appropriate corrective action.

This International Standard applies to those two-dimensional symbologies for which a reference decode algorithm has been defined, but its methodologies can be applied partially or wholly to other similar symbologies.

While this International Standard can be applied to direct part marks, it is possible that better correlation between measurement results and scanning performance will be obtained with ISO/IEC TR 29158 in combination with this International Standard.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 19762-1, *Information technology — Automatic identification and data capture (AIDC) techniques — Harmonized vocabulary — Part 1: General terms relating to AIDC*

ISO/IEC 19762-2, *Information technology — Automatic identification and data capture (AIDC) techniques — Harmonized vocabulary — Part 2: Optically readable media (ORM)*

ISO 7724-2:1984, *Paints and varnishes — Colorimetry — Part 2: Colour measurement*

ISO/IEC 15416, *Information technology — Automatic identification and data capture techniques — Bar code print quality test specification — Linear symbols*

NOTE The Bibliography lists official and industry standards containing specifications of symbologies to which (inter alia) this International Standard is applicable.

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 19762-1, ISO/IEC 19762-2, ISO/IEC 15416 and the following apply.

3.1
binarised image
binary (black/white) image created by applying the Global Threshold to the pixel values in the reference grey-scale image

3.2
effective resolution
resolution obtained on the surface of the symbol under test, normally expressed in pixels per millimetre or pixels per inch, and calculated as the resolution of the image capture element multiplied by the magnification of the optical elements of the measuring device

3.3
error correction capacity
number of codewords in a symbol (or error control block) assigned for erasure and error correction, minus the number of codewords reserved for error detection

3.4
inspection area
rectangular area which contains the entire symbol to be tested inclusive of its quiet zones

3.5
grade threshold
boundary value separating two grade levels, the value itself being taken as the lower limit of the upper grade

3.6
module error
module of which the apparent dark or light state in the binarised image is inverted from its intended state

3.7
pixel
individual light-sensitive element in an array [e.g. CCD (charge coupled device) or CMOS (complementary metal oxide semiconductor) device]

3.8
raw image
plot of the reflectance values in x and y coordinates across a two-dimensional image, representing the discrete reflectance values from each pixel of the light-sensitive array

3.9
reference grey-scale image
plot of the reflectance values in x and y coordinates across a two-dimensional image, derived from the discrete reflectance values of each pixel of the light-sensitive array by convolving the raw image with a synthesised circular aperture

3.10
reflectance margin
measurement of modulation using error correction and known module colours

3.11
sample area
area of an image contained within a circle 0,8X in diameter, X being the average module width determined by the application of the reference decode algorithm for the symbology in question or, where the application permits a range of X dimensions, the minimum module width permitted by the application specification

3.12
scan grade
result of the assessment of a single scan of a matrix symbol, derived by taking the lowest grade achieved for any measured parameter of the reference grey-scale and binarised images

4 Symbols and abbreviated terms

AN = Axial Nonuniformity

E_{cap} = error correction capacity of the symbol

e = number of erasures

FPD = Fixed Pattern Damage

GN = Grid Nonuniformity

GT = Global Threshold

MARGIN = a measure of the difference in reflectance between a module and the global threshold, the value of which goes to zero for modules of the incorrect reflectance state

MOD = an absolute measure of the difference in reflectance between a module and the global threshold

R_{max} = highest reflectance in any element or quiet zone in a scan reflectance profile, or the highest reflectance of any sample area in a two-dimensional matrix symbol

R_{min} = lowest reflectance in any element in a scan reflectance profile, or the lowest reflectance of any sample area in a two-dimensional matrix symbol

SC = Symbol Contrast (equal to R_{max} - R_{min})

t = number of errors

UEC = Unused Error Correction

5 Quality grading

5.1 General

The measurement of two-dimensional bar code symbols is designed to yield a quality grade indicating the overall quality of the symbol which can be used by producers and users of the symbol for diagnostic and process control purposes, and which is broadly predictive of the read performance to be expected of the symbol in various environments. The process requires the measurement and grading of defined parameters, from which a grade for an individual scan (scan reflectance profile grade or scan grade) is derived; the grades of multiple scans of the symbol are averaged to provide the overall symbol grade.

As a consequence of the use of different types of reading equipment under differing conditions in actual applications, the levels of quality required of two-dimensional bar code symbols to ensure an acceptable level of performance will differ. Application specifications should therefore define the required performance in terms of overall symbol grade in accordance with this standard. The guidelines in Annex D.4 are provided as an aid in writing application standards which employ this standard.

This standard defines the method of obtaining a quality grade for individual symbols. The use of this method in high volume quality control regimes may require sampling in order to achieve desired results. Such sampling plans, including required sampling rates are outside of the scope of this international standard.

NOTE Information on sampling plans may be found in the following: ISO 3951-1, ISO 3951-2, ISO 3951-3, ISO 3951-5 or ISO 2859-10.

5.2 Expression of quality grades

Although this International Standard specifies a numeric basis for expressing quality grades on a descending scale from 4 to 0, with 4 representing the highest quality, individual parameter grades and individual scan grades may also be expressed on an equivalent alphabetic scale from A to D, with a failing grade of F, in application standards with a historical link to ANSI X3.182.

Table 1 maps the alphabetic and numeric grades to each other.

Table 1 — Equivalence of numeric and alphabetic quality grades

Numeric grade	Alphabetic Grade
4	A
3	B
2	C
1	D
0	F

5.3 Overall Symbol Grade

The overall symbol grade shall be calculated as defined in 6.2.6 or 7.10. Overall symbol grades shall be expressed to one decimal place on a numeric scale ranging in descending order of quality from 4,0 to 0,0.

Where a specification defines overall symbol grades in alphabetic terms the relative mapping of the alphabetic and numeric grades is as illustrated in Figure 1 below. For example, the range of 1,5 to immediately below 2,5 corresponds to grade C.

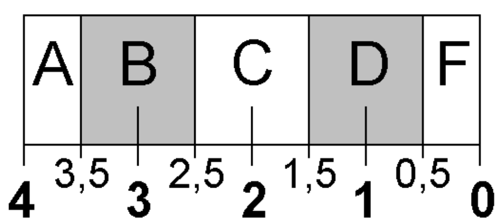


Figure 1 — Mapping of alphabetic and numeric overall symbol grades

5.4 Reporting of symbol grade

A symbol grade is only meaningful if it is reported in conjunction with the illumination and aperture used. It should be shown in the format *grade/aperture/light/angle*, where:

- "*grade"* is the overall symbol grade as defined in 6.2.6 or 7.10, i.e. the arithmetic mean to one decimal place of the scan reflectance profile or scan grades,

- *"aperture"* is the aperture reference number (from ISO/IEC 15416 for linear scanning techniques, or the diameter in thousandths of an inch (to the nearest thousandth) of the synthetic aperture defined in 7.3.3),

- "*light"* defines the illumination: a numeric value indicates the peak light wavelength in nanometres (for narrow band illumination); the alphabetic character W indicates that the symbol has been measured with broadband illumination ("white light") the spectral response characteristics of which must imperatively be defined or have their source specification clearly referenced,

- *"angle"* is an additional parameter defining the angle of incidence (relative to the plane of the symbol) of the illumination. It shall be included in the reporting of the overall symbol grade when the angle of incidence is other than 45°. Its absence indicates that the angle of incidence is 45°.

NOTE While illumination from four sides with an angle of incidence of 45° is the default, other angles of incidence may be specified as requirements for grading by specifying the angle instead of leaving it blank. Other lighting options are defined in ISO/IEC TR 29158 which may be more appropriate for direct part marking applications, especially in applications which rely on symbols marked on reflectance substrates.

An asterisk following the value for "grade", in the case of a two-dimensional matrix symbol, indicates that the surroundings of the symbol contain extremes of reflectance that may interfere with reading - see 7.6.

Examples

2,8/05/660 would indicate that the average of the grades of the scan reflectance profiles, or of the scan grades, was 2,8 when these were obtained with the use of a 0,125 mm aperture (ref. no. 05) and a 660 nm light source, incident at 45°.

2,8/10/W/30 would indicate the grade of a symbol intended to be read in broadband light, measured with light incident at 30° and using a 0,250 mm aperture (ref. no. 10), but would need to be accompanied either by a reference to the application specification defining the reference spectral characteristics used for measurement or a definition of the spectral characteristics themselves.

2,8*/10/670 would indicate the grade of a symbol measured using a 0,250 mm aperture (ref. no. 10), and a 670 nm light source, and indicates the presence of a potentially interfering extreme reflectance value in the surroundings of the symbol.

NOTE The same notation is used to specify a minimum grade that is required in an application as is a grade that is obtained by measuring a symbol in accordance with this standard. For example, an application standard may specify a symbol quality requirement as 1.5/05/660 and this would be met by a measured grade of X.X/05/660 as long as X.X is a number that is greater or equal to 1.5. However, this requirement would not be met by 2.0/10/660 nor 3.0/05/W nor 3.5/05/660/30.

6 Measurement methodology for two-dimensional multi-row bar code symbols

6.1 General

The evaluation of two-dimensional multi-row bar code symbols shall be based on the application of the methodology of ISO/IEC 15416, modified as described in 6.2.2 or 6.3, and if appropriate for the symbology, on the application of the additional provisions described in 6.2.3, 6.2.4 and 6.2.5, to derive an overall symbol grade. Ambient light levels shall be controlled in order not to have any influence on the measurement results. The symbol shall be scanned using the light wavelength(s) and effective aperture size specified in the appropriate application standard. When performing a measurement, the scan lines should be made perpendicular to the height of the bars in the start and stop characters and should as far as possible pass through the centres of rows in order to minimise the effect of cross-talk from adjacent rows. In the case of area

imaging techniques, a number of scan lines, perpendicular to the height of the bars and sufficient to cover all rows of the symbol, shall be synthesised by convolving the raw image with the appropriate synthetic aperture.

6.2 Symbologies with cross-row scanning ability

6.2.1 Basis of grading

The distinguishing feature of these symbologies is their ability to be read with scan lines that cross row boundaries. Symbologies of this type, at the date of publication of this International Standard, also share the feature that the start and stop patterns (or equivalent features of the symbol, e.g. the Row Address Patterns of MicroPDF417) are constant from row to row, or the position of only one edge in these patterns varies by no more than 1X in adjacent rows of the symbol. These symbologies shall be graded in respect of:

— Analysis of the scan reflectance profile (based on ISO/IEC 15416) (see 6.2.2)

— Codeword Yield (see 6.2.3)

— Unused Error Correction (see 6.2.4)

— Codeword print quality (see 6.2.5)

6.2.2 Grade based on analysis of scan reflectance profile

The start and stop or equivalent (e.g. Row Address) patterns of the symbol shall be evaluated according to ISO/IEC 15416. Regions with data content will be evaluated separately as described in 6.1.2, 6.1.3 and 6.1.4. Test scans of the Start and Stop patterns shall be graded using all parameters specified in ISO/IEC 15416.The effective aperture size is specified in the appropriate application standard or is the default aperture size appropriate for the symbol X dimension given in ISO/IEC 15416.

For the analysis of the scan reflectance profiles, the number of scans should be ten, or the height of the symbol divided by the measuring aperture if this quotient is less than ten. Scans should be approximately evenly spaced over the height of the symbol. For example, in a twenty-row symbol the ten scans might be performed in alternate rows. In a two-row symbol, up to five scans might be performed in each row, at different positions in the height of the bars. The symbology specification may give more specific guidance on the selection of the scans to be used.

To identify bars and spaces, a Global Threshold for each scan has to be determined. Global Threshold shall be equal in reflectance to $(R_{max} + R_{min}) / 2$, where the values R_{max} and R_{min} are respectively the highest and the lowest reflectances in the scan. All regions above the Global Threshold shall be considered spaces (or quiet zones) and all regions below shall be considered bars.

Edge locations shall be determined as the points where the reflectance value is midway between the highest reflectance in the adjoining space and the lowest reflectance in the adjoining bar, in accordance with ISO/IEC 15416.

For the evaluation of the parameters 'decode' and 'decodability' the reference decode algorithm for the symbology shall be applied.

Each scan shall be graded as the lowest grade for any individual parameter in that scan. The grade based on scan reflectance profiles shall be the arithmetic mean of the grades for the individual scans.

The measurement of bar width gain or loss may be used for process control purposes. Note that this method will not be sensitive to printing variations parallel to the height of the start and stop characters. If a full analysis of the printing process is desired, symbols should be printed and tested in both orientations.

6.2.3 Grade based on Codeword Yield

This parameter measures the efficiency with which linear scans can recover data from a two-dimensional multi-row symbol. The Codeword Yield is the number of validly decoded codewords expressed as a percentage of the maximum number of codewords that could have been decoded (after adjusting for tilt). A poor Codeword Yield, for a symbol whose other measurements are good, may indicate a Y-axis print quality problem (such as those shown in Table C.1).

Obtain a matrix of the correct symbol character values, such as would result from successful completion of the *UEC* calculations (see 6.2.4). This matrix is used as the "final decode of the symbol" used in subsequent steps to determine validly decoded codewords.

An individual scan qualifies for inclusion in the Codeword Yield calculation if it meets either of two conditions:

1) The scan did not include recognised portions of either the top or the bottom row of the symbol. At least one of the Start or Stop (or Row Address) patterns shall have been successfully decoded from that scan, together with at least one additional codeword or the corresponding second Start or Stop pattern, or Row Address Pattern.

2) The scan included recognised portions of either the top or the bottom row of the symbol. Both the Start and Stop patterns of the symbol shall have been successfully decoded from that scan.

It is important to note that an extension to the symbology's Reference Decode Algorithm is required, in order to detect and decode a pair of Start and Stop patterns when neither of the adjacent codewords is decodable. As examples, a linear search for a matching pair of PDF417 Start and Stop patterns, or a linear search for a matching pair of MicroPDF417 Row Indicator Patterns, would fulfil this requirement for scans where the Reference Decode Algorithm alone did not decode both patterns; thus this extension can qualify a scan where no codewords (other than the matched end patterns) were decoded. Note however, that a scan that contains only a *single* decoded Start or Stop pattern found by this linear search does not count as a qualified scan, if no other codewords or corresponding second Start or Stop pattern, or Row Address Pattern, were also decoded.

Decode the symbol completely and populate the symbol matrix.

For each qualified scan, compare the codewords actually decoded with the codewords in the symbol matrix and count the number of codewords that match. Accumulate the total number of validly decoded codewords, and update a count of the number of times each codeword of the symbol has been decoded and a count of the number of times each row has been detected. Also record a count of the number of detected row crossings in each scan (a crossing is “detected” when a scan line yields correctly-decoded codewords from adjacent rows).

After processing each scan, calculate the maximum number of codewords that could have been decoded thus far, as the number of qualified scans multiplied by the number of columns in the symbol (excluding the fixed patterns, such as the Start and Stop patterns of PDF417 or the Row Address Indicators of MicroPDF417).

The entire symbol shall be scanned multiple times until three conditions are met:

1) the maximum number of codewords that could have been decoded is at least ten times the number of codewords in the symbol,

2) the highest and lowest decodable rows (which may not necessarily be the first and last rows) of the symbol have each been scanned at least three times, and

3) at least ($0.9n$) of the codewords (data or error correction) have been successfully decoded two or more times, where n is the number of non-error-correction data codewords in the symbol.

EXAMPLE Taking a PDF417 symbol with 6 rows and 16 columns and error correction level 4, the total number of codewords is 96, of which 64 are data and 32 error correction. To fulfil condition 1, the maximum number of codewords that could have been decoded must be at least 960. To fulfil condition 3, since n is 64, at least 58 of the codewords must have been decoded twice or more ($0,9 \times 64 = 57,6$).

If the ratio of the total number of validly decoded codewords to the total number of detected row crossings is less than 10 : 1, then discard the measurements just obtained, and repeat the measurement process, adjusting the tilt angle of the scan line to reduce the number of row crossings.

Otherwise, to compensate for any residual tilt, subtract the number of detected row crossings from the calculated maximum number of codewords that could have been decoded.

Codeword Yield shall be graded as shown in Table 2:

Table 2 — Grading of Codeword Yield

Codeword Yield	Grade
≥ 71%	4
≥ 64%	3
≥ 57%	2
≥ 50%	1
< 50%	0

6.2.4 Grade based on unused error correction

Decode the symbol completely and process scans until the number of decoded codewords stabilises. Calculate the unused error correction (*UEC*) as $UEC = 1{,}0 - ((e + 2t) / E_{cap})$, where e = the number of erasures, t = the number of errors and E_{cap} = the error correction capacity of the symbol (the number of error correction codewords minus the number of error correction codewords reserved for error detection). If no error correction has been applied to the symbol, and if the symbol decodes, $UEC = 1$. If $(e + 2t)$ is greater than E_{cap}, $UEC = 0$. In symbols with more than one (e.g. interleaved) error correction block, *UEC* shall be calculated for each block independently and the lowest value shall be used for grading purposes.

Unused Error Correction shall be graded as shown in Table 3:

Table 3 —Grading of Unused Error Correction

Unused Error Correction	Grade
≥ 0,62	4
≥ 0,50	3
≥ 0,37	2
≥ 0,25	1
< 0,25	0

6.2.5 Grade based on codeword print quality

The approach detailed in this subclause provides additional diagnostic information and enables allowance to be made for the effect of error correction in masking less than perfect attributes of the symbol that influence symbol quality, by applying an overlay technique as described in Annex F. It enables the Decodability, Defects and Modulation parameters of scan reflectance profiles covering the entire data region of the symbol to be graded in accordance with ISO/IEC 15416.

This approach uses the following procedure for the assessment of each of the three parameters. In symbols with more than one (e.g. interleaved) error correction block, it shall be applied to each block independently and the lowest value shall be used for grading purposes.

The entire symbol shall be scanned until 0.9n codewords (where n has the same meaning as in 6.2.3) have been decoded ten times or until it is certain that each codeword has been scanned at least once without inter-row interference. In each scan, the Decodability, Defects and Modulation parameters shall be measured in each symbol character in accordance with ISO/IEC 15416. The calculation of all three parameters shall be based on the value of Symbol Contrast obtained from R_{max} and R_{min} in that scan line. The interim codeword grade of each parameter (Modulation, Defects and Decodability) for each codeword is the highest codeword grade for that parameter obtained on any scan for that codeword.

Where the rows include overhead characters (other than the Start and Stop, or equivalent patterns), for example Row Indicators in PDF417 symbols, that are not included in the error correction calculation, these overhead characters shall be assessed first for each row together with the corresponding characters from the rows immediately above and below the row being considered. The highest interim codeword grade for any of these six (or four, in the case of the top or bottom row) characters shall be the overhead grade used to moderate the interim codeword grades for the codewords in the row. If a data codeword's interim codeword grade is higher than the grade obtained by the overhead characters, the data codeword's interim codeword grade shall be reduced to the overhead grade. The interim parameter grades so obtained shall then be modified to allow for the influence of error correction, as described below.

For each parameter, the cumulative number of symbol characters achieving each grade from 4 to 0 or a higher grade, and those not decoded, shall be counted, and the counts shall be compared with the error correction capacity of the symbol as follows:

For each grade level, assuming that all symbol characters not achieving that grade or a higher grade are erasures, derive a notional grade for Unused Error Correction as described in 6.2.4, based on the percentage thresholds shown in Table 3. The codeword parameter grade shall be the lower of the grade level and the notional UEC grade.

NOTE 1 This notional grade is not related to, and does not affect, the UEC grade for the symbol as calculated according to 6.2.4, but is a means of compensating for the extent to which error correction can mask imperfections in a symbol. If one symbol has higher error correction capacity than another symbol, then the former symbol can tolerate a greater number of codewords with poor values for the parameter in question than the latter. See Annex F for a fuller description of the approach. The final codeword parameter grade for the symbol shall be the highest codeword interim grade for all grade levels.

Table 4 shows an example of grading one parameter in a symbol containing 100 symbol characters (codewords) with an error correction capacity of 32 codewords. The 100 codewords consist of 68 data codewords, 3 error correction codewords reserved for error detection, and 29 error correction codewords to be used for correcting erasures or errors, giving an erasure correction capacity of 29. The symbol would be graded 1 for the parameter concerned (the highest value in the right-hand column).

NOTE 2 A similar calculation is performed for each of the parameters Modulation, Defects and Decodability

Table 4 — Example of codeword print quality parameter grading in symbols with cross-row scanning ability, applying overlay procedure in Annex F

***MOD*/Defects/Decodability grade level (a)**	**No. of codewords at level a**	**Cumulative no. of codewords at level a or higher (b)**	**Remaining codewords (treated as erasures) (100 - b) (c)**	**Notional unused error correction capacity (29 – c)**	**Notional UEC (%)**	**Notional UEC grade (d)**	**Codeword interim grade level (Lower of a or d) (e)**
4	40	40	60	(exceeded)	<0	0	0
3	20	60	40	(exceeded)	<0	0	0
2	10	70	30	(exceeded)	<0	0	0
1	10	80	20	9	31%	1	1
0	7	87	13	16	55%	3	0
Not decoded	13	100					
					Parameter grade (Highest value of **e**)		1

6.2.6 Overall symbol grade

The overall symbol grade shall be the lowest of the grade based on analysis of the scan reflectance profile in accordance with 6.2.2, and the grades based on Codeword Yield, Unused Error Correction and codeword print quality in accordance with 6.2.3, 6.2.4 and 6.2.5.

The flowchart in Figure 2 summarises the process.

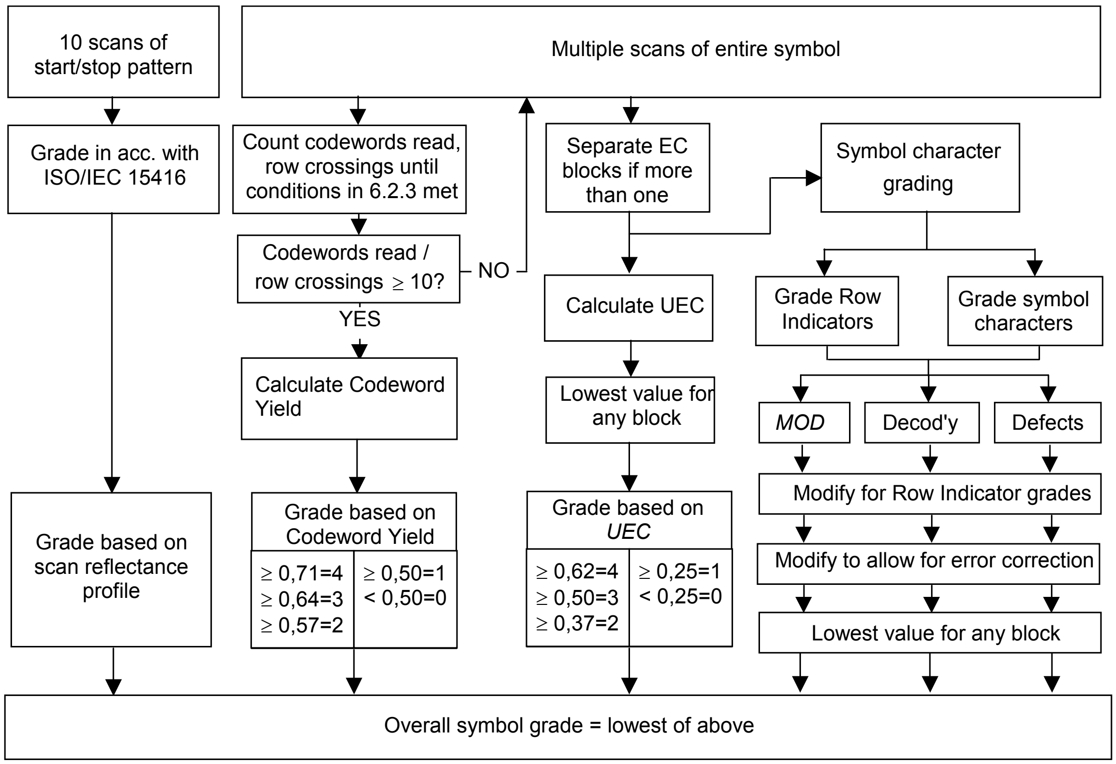


Figure 2 — Grading process for multi-row symbols with cross-row scanning ability

6.3 Symbologies requiring row-by-row scanning

The distinguishing feature of these symbologies is that they require a scan line to traverse a complete row from start to stop pattern (or in the reverse direction) without crossing into an adjacent row and that they require all rows to be scanned.

Each row shall be evaluated in accordance with ISO/IEC 15416 as though it were a separate symbol. Scan lines shall pass through the inspection band of the central 80% of the height of each row, as specified in ISO/IEC 15416, in order to minimise the effects of cross-talk from adjacent rows. The number of scans per row should be the lower of ten, or the inspection band height divided by the aperture diameter. The overall symbol grade shall be the lowest overall grade obtained for any row.

7 Measurement methodology for two-dimensional matrix symbols

7.1 Overview of methodology

The measurement methodology defined in this clause is designed to maximize the consistency of both reflectivity and dimensional measurements of symbols on various substrates. The basis of this methodology is the measurement of reflectance from the symbol. This methodology is also intended to correlate with conditions encountered in two-dimensional matrix scanning systems.

The method starts by obtaining the raw image, which is a high-resolution grey-scale image of the symbol captured under controlled illumination and viewing conditions. The stored raw image is then converted into a

reference grey-scale image, by convolving the raw image with a synthetic circular aperture. From the reference grey-scale image, the Symbol Contrast, Modulation and Fixed Pattern Damage parameters are measured and graded. A secondary binarised image is produced from the reference grey-scale image by applying a Global Threshold, and this binarised image is then analysed and graded for the parameters of Decode, Axial Nonuniformity, Grid Nonuniformity, and Unused Error Correction, together with any additional parameters defined in the symbology or application specification. The methodology recognises possible extreme reflectance values in the neighbourhood of the symbol, which might interfere with reading; however, only their presence is indicated in the report of the overall symbol grade.

In addition, print growth or loss is measured along each axis of the symbol and reported as an ungraded process control measurement.

The scan grade is the lowest grade achieved for these seven parameters and any others specified for a given symbology or application.

7.2 Obtaining the test images

7.2.1 Measurement conditions

A test image of the symbol shall be obtained in a configuration that mimics the typical scanning situation for that symbol, but with substantially higher resolution (see 7.3.3), uniform illumination, and at best focus. The reference optical arrangement is defined in 7.3.4 and should be used where application requirements do not call for a specialised optical arrangement; alternative optical arrangements (two of which are defined in 7.3.4) may be used provided that the measurements obtained with them can be correlated with the use of the reference optical arrangement.

Measurements shall be made with light of a single peak wavelength or set of spectral characteristics and a known diameter of measuring aperture, both of which shall be defined by the application specification or determined in accordance with 7.3.2 and 7.3.3. Ambient light levels shall be controlled in order not to have any influence on the measurement results.

Whenever possible, measurements shall be made on the symbol in its final configuration, i.e. the configuration in which it is intended to be scanned. The measurement method is described in 7.6 and 7.7, and Annex B, and is intended to prevent extreme reflectance values outside the symbol area (e.g. when surrounded by free air or a highly specularly reflective surface) from distorting the symbol contrast measurements.

Specialized applications (e.g. the measurement of quality of symbols produced by engraving or etching the substrate surface) clearly must dictate the colour and angle of symbol illumination as well as the required imaging resolution, but the general test set-up defined in 7.3.4 should work suitably for many open applications. For Direct Part Mark applications, a modified version of the methodology described in this standard may be more appropriate. The modified methodology is formally defined in ISO/IEC TR 29158 and may be followed if such is in accordance with the relevant application standard.

Two principles govern the design of the optical set-up. First, the test image's grey-scale shall be nominally linear and not be enhanced in any way. Second, the image resolution shall be adequate to produce consistent readings. See 7.3.3.

7.2.2 Raw image

The raw image is a plot of the actual reflectance values for each pixel of the light-sensitive array, from which are derived the reference grey-scale image and the binarised image which are evaluated for the assessment of symbol quality.

7.2.3 Reference grey-scale image

The reference grey-scale image is obtained from the raw image by processing the individual pixel reflectance values through a synthetic circular aperture as defined in 7.3.3. It is used for the assessment of the parameters Symbol Contrast, Modulation, Reflectance Margin and Fixed Pattern Damage.

7.2.4 Binarised image

The binarised image is obtained from the reference grey-scale image by applying a Global Threshold midway between R_{max} and R_{min}, determined as defined in 7.6. It is used for the assessment of the parameters Decode, Axial Non-uniformity, Grid Non-uniformity, and Unused Error Correction.

7.3 Reference reflectivity measurements

7.3.1 General requirements

Equipment for assessing the quality of symbols in accordance with this clause shall comprise a means of measuring and analysing the variations in the reflectivity of a symbol on its substrate over an inspection area which shall cover the full height and width of the symbol including all quiet zones.

All measurements on a two-dimensional matrix symbol shall be made within the inspection area defined in accordance with 7.3.5.

The measured reflectance values shall be expressed in percentage terms either with reference to the reflectance of a barium sulphate or magnesium oxide reference sample complying with the requirements of ISO 7724-2, which shall be taken as 100 per cent, or by means of calibration and reference to recognised national standards laboratories.

7.3.2 Light source

The peak light wavelength or, in the case of applications designed for the use of broadband illumination, the reference spectral response characteristics, should be specified in the application specification to suit the intended scanning environment. When the peak wavelength or spectral characteristics are not specified in the application specification, measurements should be made using light of characteristics that approximate most closely to those expected to be used in the scanning process. Light sources may either have inherently narrow band or near-monochromatic characteristics or have broad bandwidths; in the latter case the spectral response of the measuring system may be restricted to the desired peak wavelength(s) by the interposition of an appropriate narrow band filter in the optical path.

NOTE Special care is necessary when making measurements with broadband illumination. The overall spectral response of the measurement and reading systems must be defined and matched in order to make accurate and repeatable measurements of the grey-scale reflectance of a sample area that correlate with the intended system. Overall spectral response includes the spectral distribution of the light source, the response of the detector and any associated filter characteristics.

Refer to Annex D for guidance on the selection of the light source.

7.3.3 Effective resolution and measuring aperture

The measuring aperture is specified by the user application specification to suit the *X* dimension of the symbol and the intended scanning environment. Matrix symbol grading shall be carried out using a synthesised aperture. An aperture size in the range of 50% to 80% of the smallest *X* dimension to be encountered in an application is recommended. In an application where symbols of differing *X* dimensions will be encountered, the application standard should ensure that all measurements are made with the aperture appropriate to the smallest *X* dimension to be encountered. See Annex D.2 for guidance in writing application standards and considering the tradeoffs associated with the choice of aperture size.

The effective resolution of an instrument that implements this international standard shall be sufficient to ensure that the parameter grading results are consistent irrespective of the rotation of the symbol. The effective resolution is the product of the resolution of the light-sensitive array and of the magnification of the associated optical system and effected by distortions introduced by the optical system. The reference optical arrangement requires high resolution, such as an effective resolution of not less than ten pixels per module in width and height.

NOTE Implementations (e.g. commercial verifiers) may use fewer pixels per module, providing that the consistency irrespective of rotation mentioned above is achieved on the test symbols specified in ISO/IEC 15426-2.

7.3.4 Optical geometry

A reference optical geometry is defined for reflectivity measurements and consists of:

— flood incident illumination, uniform across the inspection area, from a set of four light sources arranged at 90 degree intervals around a circle concentric with the inspection area and in a plane parallel to that of the inspection area, at a height which will allow incident light to fall on the centre of the inspection area at an angle of 45° to its plane, and

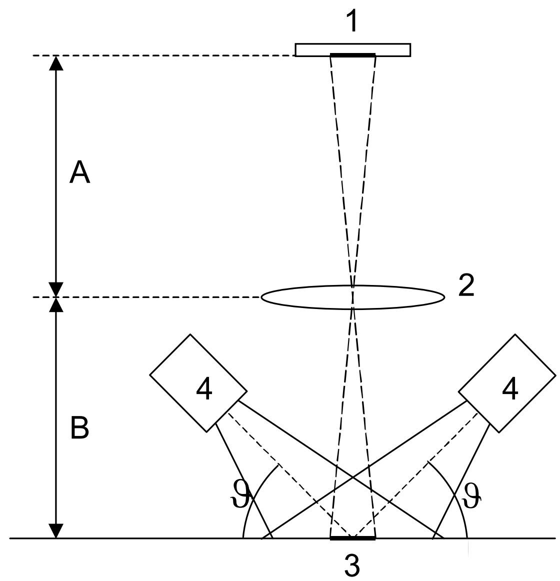
— a light collection device, the optical axis of which is perpendicular to the inspection area and passes through its centre, and which focuses an image of the test symbol on a light-sensitive array.

The light reflected from the inspection area (see 7.3.5) plus the 20*Z* extension defined in 7.7 shall be collected and focussed on the light-sensitive array.

Implementations may use alternative optical geometries and components, provided that their performance can be correlated with that of the reference optical arrangement defined in this section. Figures 3 and 4 illustrate the principle of the optical arrangement, but are not intended to represent actual devices; in particular the magnification of the device is likely to differ from 1:1. In addition, many devices include filters to modify the spectral characteristics or restrict the effect of unwanted spectral components. Implementations should have sufficient resolution irrespective of the rotation as stated in 7.3.3, unless the manufacturer defines handling instructions which restricts the angle of the symbol in relation to the camera chip orientation.

This reference geometry is intended to provide a basis to assist the consistency of measurement and may not correspond with the optical geometry of individual scanning systems. As stated in 7.2, specialised applications, and especially those involving direct part marking which employs physical changes to the surface of the substrate for the creation of the graphic image, may require the angle of illumination, in particular, to be set to a different particular angle such as 30° to the plane of the symbol. If an angle other than the default is specified in the application specification, then the angle of incidence of the light shall be stated as a fourth parameter when reporting the overall symbol grade, as described in 5.4.

The modified methodology defined in ISO/IEC TR 29158 intended for direct part marking applications defines more illumination options including diffuse illumination at a near-90° incident angle.



1 – Light sensing element

2 – Lens providing 1:1 magnification (measurement *A* = measurement *B*)

3 – Inspection area

4 – Light sources

ϑ - Angle of incidence of light relative to plane of symbol (default = 45°, optionally 30° or 90° diffuse)

Figure 3 — Reference optical arrangement – side view

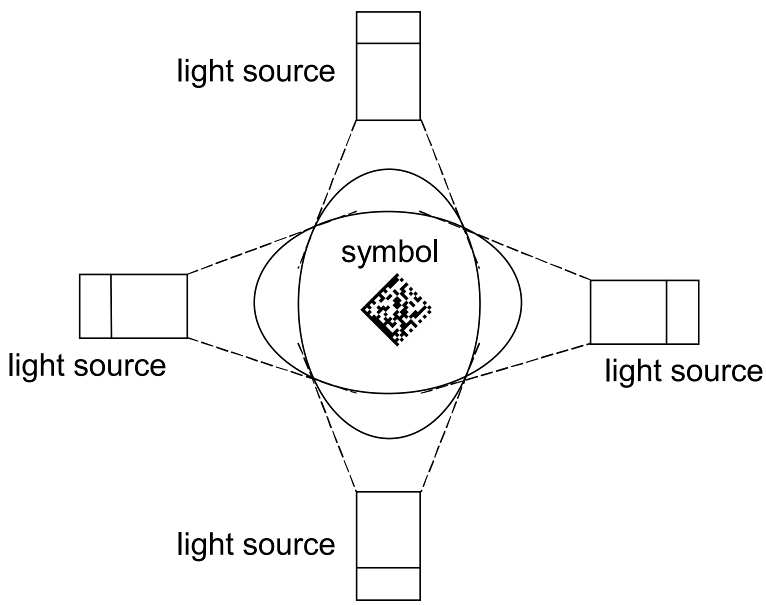


Figure 4 — Reference optical arrangement – plan view

7.3.5 Inspection area

The area within which all measurements shall be made shall be a rectangular area framing the complete symbol, including quiet zones. The centre of the inspection area shall be as close as practicable to the centre of the field of view.

NOTE The inspection area is not the same as the field of view of the verifier, which should be sufficiently large to include the whole symbol plus the 20*Z* extension described in 7.7.

7.4 Number of scans

The overall symbol grade is obtained through one measurement, with the symbol oriented in any rotation with respect to the measuring device, in a plane perpendicular to the optical axis to the imager sensor. See D.5 for information regarding the fact that this international standard previously required five scans at different rotations to be made and averaged to obtain an overall grade.

Note: This may not be appropriate for symbols on certain substrates or marking methods that do not exhibit uniform diffuse reflection and therefore exhibit variations in symbol reflectance characteristics when viewed in different orientations relative to the axis of the measuring device. Such symbols may be more appropriately measured by following the modified methodology in ISO/IEC TR 29158.

7.5 Basis of scan grading

Two-dimensional symbol quality assessment shall be based on the measurement and grading of parameters of the reference grey-scale image, the binarised image derived from it, and the application of the reference decode algorithm to these, as defined in 7.8. Quality grading of these parameters shall be used to provide a relative measure of symbol quality under the measurement conditions used. Each parameter shall be measured and a grade on a descending scale of integers from 4 to 0 shall be allocated to it. The grade 4 represents the highest quality, while the grade 0 represents failure.

7.6 Grading procedure

A flowchart illustrating the procedure is shown in Annex B.

Centre the symbol in the field of view

Capture the raw image (see 7.2.2).

Find and replace the brightest .005% pixels in the overall image with the median of the nine pixels consisting of itself and its eight immediate neighbours.

Apply the aperture defined in 7.3.3 to the raw image to create a reference grey-scale image (see 7.2.3).

A circular area with a diameter 20 times the aperture diameter, centred in the reference grey-scale image, should be used to find the initial values for R_{min} and R_{max}. Using these values, determine an initial Global Threshold, create a binarised image (see 7.2.4), find the symbol and perform an initial decode.

Once the symbol has been decoded, measure revised R_{min} and R_{max} and recalculate the Global Threshold based on the whole inspection area of the reference grey-scale image (including the quiet zone). These values are used to recalculate module centres. Create a new binarised image. Perform a definitive decode and calculate all of the graded parameters of the symbol. Based on these, determine the scan grade for that image.

7.7 Additional reflectance check over extended area

If the scan grade for each of modulation, decode, and finder pattern damage is 1 or higher, then perform an additional reflectance check as follows:

> Measure R_{min} and R_{max} over an area extending to 20Z beyond the quiet zone on all sides. The field of view must be large enough to contain all points in the extended area.
>
> If either the extended-area R_{min} is lower than the revised R_{min}, or the extended-area R_{max} is higher than the revised R_{max}, then repeat the measurement of modulation and finder pattern damage. If that measurement results in a modulation or finder pattern damage grade of 0, then an asterisk is appended to the overall symbol grade. This asterisk indicates that the substrate surrounding the symbol contains extremes of reflectance that may interfere with reading.

NOTE This additional reflectance check does not alter the reported overall symbol grade nor the reported grades for the symbol contrast, modulation or finder pattern damage parameters.

The additional reflectance check may be omitted, if specifically permitted by the application specification, where the conditions under which the symbol is produced and applied are such that the risk of excessively high or low reflectance values in the extended area is insignificant, and the verifier field of view may then include only the symbol and its associated quiet zones.

7.8 Image assessment parameters and grading

7.8.1 Use of reference decode algorithm

The symbology reference decode algorithm found in the symbology specification is to be used in the verification process. In order to simplify processing the reference decode algorithm may be modified in the verifier by assuming that the symbol to be verified is approximately centred in the field of view of the device. No modifications to the reference decode algorithm that alter the functions listed below (since the adaptive grid mapping are essential to the grading process defined herein) are to be made. The reference decode performs five tasks needed for subsequent measurement of the symbol quality parameters.

- It locates and defines the area covered by the test symbol in the image.
- It determines reference points from the fixed patterns of the symbol to be used in constructing an ideal grid for measuring GNU.
- It adaptively creates a grid mapping of the data module nominal centres so as to sample them.
- It determines the nominal grid centre spacings in each axis of the symbol (the symbol X dimension)
- It performs error correction, detecting if symbol damage has consumed any of the error budget.
- It attempts to decode the symbol.

These functions each facilitate one or more of the measurements described in the following subclauses.

The image parameters described in 7.8.2 to 7.8.9 shall be assessed for compliance with this standard.

7.8.2 Decode

The Decode parameter tests, on a Pass/Fail basis, whether the symbol has all its features sufficiently correct to be readable by the reference decode algorithm.

The symbology reference decode algorithm shall be used to decode the symbol using the module centre positions on the grid determined by processing the binarised image.

If the image cannot be decoded using the symbology reference decode algorithm, then it shall receive the failing grade 0. Otherwise, it shall receive the grade 4.

7.8.3 Symbol Contrast

Symbol Contrast tests that the two reflective states in the symbol, namely light and dark, are sufficiently distinct within the symbol.

Using the reference grey-scale image of the symbol, measure the highest and lowest reflectance values in the inspection area. Symbol contrast is the difference between the highest and lowest reflectance values in the inspection area. The reflectance values to be used are the revised R_{max} and R_{min} as defined in 7.6.

$$SC = R_{max} - R_{min}$$

Symbol contrast shall be graded as shown in Table 5.

Table 5 — Symbol Contrast grading

Symbol Contrast	Grade
≥ 70%	4
≥ 55%	3
≥ 40%	2
≥ 20%	1
< 20%	0

7.8.4 Modulation and related measurements

7.8.4.1 Modulation

Modulation is a measure of the uniformity of reflectance of the dark and light modules respectively. Factors such as print growth (or loss), misplacement of a module relative to the grid intersection, the optical characteristics of the substrate and uneven printing may reduce the absolute value of the difference between the reflectance of a module and the Global Threshold. A low Modulation may increase the probability of a module being incorrectly identified as dark or light.

The reflectance value of each module in the symbol shall be measured by superimposing on the reference grey-scale image the grid determined by applying the symbology reference decode algorithm to the binarised image. Calculate MOD, the Modulation value of each module as follows:

$$MOD = 2 * (\text{abs}\,(R - GT)) / SC$$

Where MOD = modulation
R is the reflectance of the module
GT is the Global Threshold
SC is the Symbol Contrast

Assign the grade level for each module according to Table 6. For each codeword, select the minimum modulation grade of all modules in the codeword. As suggested by the absolute value in the function for MOD, whether a codeword is decoded correctly has no bearing on the grade level that is assigned. In this way, Modulation differs from Reflectance Margin, see 7.8.4.3.

Table 6 — Module grading for Modulation and Reflectance Margin

MOD or MARGIN	Module Grade
≥ 0,50	4
≥ 0,40	3
≥ 0,30	2
≥ 0,20	1
< 0,20	0

The cumulative number of codewords achieving each grade shall be counted and compared with the error correction capacity of the symbol as follows:

For each grade level, assuming that all codewords not achieving that grade or a higher grade are errors, derive a notional Unused Error Correction grade as described 7.8.8. Take the lower of the grade level and the notional UEC grade.

NOTE This notional grade is not related to, and does not affect, the *UEC* grade for the symbol as calculated according to 7.8.8, but is a means of compensating for the extent to which error correction can mask imperfections in a symbol. If one symbol has higher error correction capacity than another symbol, then the former symbol can tolerate a greater number of codewords with low modulation than the latter. See Annex F for a fuller description of the approach.

Then the Modulation grade for the symbol shall be the highest of the resulting values for all grade levels. When the symbol consists of more than one (e.g. interleaved) error correction block, each block shall be assessed independently and the lowest grade for any block shall be taken as the Modulation grade of the symbol.

Table 7(A) shows an example of grading Modulation in a symbol containing 120 codewords, 60 of which are error correction codewords with a capacity to correct up to 30 errors in a single error correction block. Modulation grade of the symbol in the example would be 2 (the highest value in the right-hand column).

Table 7(A) — Example of Modulation grading in a two-dimensional matrix symbol

MOD codeword grade level (**a**)	No. of codewords at level **a**	Cumulative no. of codewords at level **a** or higher (**b**)	Remaining codewords (treated as errors) (120 - **b**) (**c**)	Notional unused error correction capacity (30 – **c**)	Notional *UEC* (%)	Notional *UEC* grade (**d**)	Lower of **a** or **d** (**e**)
4	25	25	95	(exceeded)	<0	0	0
3	75	100	20	10	33,3%	1	1
2	15	115	5	25	83,3%	4	**2**
1	3	118	2	28	93.3%	4	1
0	2	120	0	30	100%	4	0
					Modulation grade (Highest value of **e**):		2

In this example, some codewords may contain errors but that does not affect the calculation.

7.8.4.2 Contrast Uniformity

Contrast Uniformity is an optional parameter that can be a useful process control tool for measuring localized contrast variations. Contrast Uniformity does not affect the overall grade.

Contrast Uniformity is defined as the minimum MOD value found in any module contained in the data region of the symbol in clause 7.8.4.1.

7.8.4.3 Reflectance Margin

Reflectance Margin is a measure of how well each module is correctly distinguishable as light or dark in comparison to the global threshold. Factors such as print growth (or loss), misplacement of a module relative to the grid intersection, the optical characteristics of the substrate, uneven printing, or encodation errors, may reduce or even eliminate the margin for error between the reflectance of a module and the Global Threshold. A low Reflectance Margin may increase the probability of a module being incorrectly identified as dark or light.

The reflectance value of each module in each codeword in the symbol shall be measured by superimposing on the reference grey-scale image the grid determined by applying the symbology reference decode algorithm to the binarised image.

Since the correct state of each module is known after decoding, any modules which are decoded incorrectly are assigned a *MARGIN* value of 0.

For modules whose correct state is light:

$$MARGIN = 2 * (R - GT) / SC \text{ for } R \geq GT$$

$$MARGIN = 0 \text{ for } R < GT$$

and for modules whose correct state is dark:

$$MARGIN = 2 * (GT - R) / SC \text{ for } R < GT$$

$$MARGIN = 0 \text{ for } R \geq GT$$

Where *MARGIN* = the reflectance margin of the module
R is the reflectance of the module
GT is the Global Threshold
SC is the Symbol Contrast

Assign the grade level for each module according to Table 6. For each codeword, select the minimum grade for *MARGIN* of all modules in the codeword. Since codewords which are misdecoded are given grade level of 0, Reflectance Margin differs from Modulation, see 7.8.4.1.

The cumulative number of codewords achieving each grade shall be counted and compared with the error correction capacity of the symbol as follows:

For each grade level, assuming that all codewords not achieving that grade or a higher grade are errors, derive a notional Unused Error Correction grade as described in 7.8.8. Take the lower of the grade level and the notional UEC grade.

NOTE This notional grade is not related to, and does not affect, the *UEC* grade for the symbol as calculated according to 7.8.8, but is a means of compensating for the extent to which error correction can mask imperfections in a symbol. If one symbol has higher error correction capacity than another symbol, then the former symbol can tolerate a greater number of codewords with low modulation than the latter. See Annex F for a fuller description of the approach.

Then the Reflectance Margin grade for the symbol shall be the highest of the resulting values for all grade levels.

Tabl**e 7(B)** shows an example of grading Reflectance Margin in a symbol containing 120 codewords, 60 of which are error correction codewords with a capacity to correct up to 30 errors in a single error correction block. The Modulation grade of the symbol in the example would be 1 (the highest value in the right-hand column).

Table 7(B)— Example of Reflectance Margin grading in a two-dimensional matrix symbol, applying overlay procedure in Annex F

***MARGIN* codeword grade level (a)**	**No. of codewords at level a**	**Cumulative no. of codewords at level a or higher (b)**	**Remaining codewords (treated as errors) (120 - b) (c)**	**Notional unused error correction capacity (30 - c)**	**Notional *UEC* (%)**	**Notional *UEC* grade (d)**	**Lower of a or d (e)**
4	15	15	105	(exceeded)	<0	0	0
3	70	85	35	(exceeded)	<0	0	0
2	15	100	20	10	33,3%	1	**1**
1	5	105	15	15	50%	3	**1**
0	15	120	0	30	100%	4	0
					Reflectance Margin grade (Highest value of **e)**:		1

This example represents values from the same symbol used in Table 7(A). However, in this example ten codewords from level 4, and five codewords from level 3 are detected to contain at least one module which is on the wrong side of the global threshold and are therefore errors. These codewords are therefore counted at level 0 in this example. The resulting grade too is changed significantly.

7.8.5 Fixed Pattern Damage

This parameter tests that damage to the finder pattern, quiet zone, timing, navigation and other fixed patterns in a symbol does not reduce unacceptably the ability of the reference decode algorithm to locate and identify the symbol within the field of view, by inverting the apparent state of one or more modules from light to dark or vice versa. The particular patterns to be considered, and the amounts of damage corresponding to the various grade thresholds, require to be specified independently for the symbology concerned.

Fixed Pattern Damage is evaluated in the reference grey-scale image in terms of the number of module errors (i.e. modules that appear as the inverse of the intended colour) in the feature (or part of the feature) concerned. Where the symbol comprises a number of distinct features (e.g. finder pattern, timing pattern) each feature may require to be evaluated separately and the worst value used for grading purposes.

Fixed Pattern Damage shall be graded using the threshold values appropriate to each symbology, specified in Annex A, or in the symbology specification, the latter taking precedence.

7.8.6 Axial Nonuniformity

Two-dimensional matrix symbols include data fields of modules nominally lying in a regular polygonal grid, and any reference decode algorithm must adaptively map the centre positions of those modules to extract the data. Axial Nonuniformity measures and grades the spacing of the mapping centres, i.e. the sampling points, or intersections of the grid obtained by applying the reference decode algorithm to the binarised image, in the direction of each of the grid's major axes. Axial Nonuniformity tests for uneven scaling of the symbol which would hinder readability at some non-normal viewing angles more than at others.

The spacings between adjacent sampling points are independently sorted for each polygonal axis, then the average spacings X_{AVG}, Y_{AVG}, … along each axis are computed. Axial Nonuniformity is a measure of how much the sampling point spacing differs from one axis to another, namely:

$$AN = \text{abs}(X_{AVG} - Y_{AVG}) / ((X_{AVG} + Y_{AVG}) / 2)$$

where abs() yields the absolute value. If a symbology has more than two major axes, then Axial Nonuniformity is computed for those two average spacings which differ the most.

Axial Nonuniformity shall be graded as shown in Table 8.

Table 8 — Axial Nonuniformity grading

Axial Nonuniformity	Grade
≤ 0,06	4
≤ 0,08	3
≤ 0,10	2
≤ 0,12	1
> 0,12	0

7.8.7 Grid Nonuniformity

Grid Nonuniformity measures and grades the largest vector deviation of the grid intersections, determined by the reference decode algorithm from the binarised image of a given symbol, from their ideal theoretical position.

Using the reference decode algorithm for the symbology, plot the positions of all grid intersections in the data area of the symbol and compare these positions with the ideal grid in a theoretical perfect symbol of the same nominal dimensions. The greatest distance between the actual and the theoretical position of any intersection, expressed as a fraction of the *X* dimension of the symbol, shall be taken for grading purposes.

The theoretical grid shall be constructed by equal spacing from the minimum number of reference points defined by the reference decode algorithm from the fixed patterns in the symbol.

Grid Nonuniformity shall be graded as shown in Table 9.

Table 9 — Grid Nonuniformity grading

Grid Nonuniformity	Grade
≤ 0,38	4
≤ 0,50	3
≤ 0,63	2
≤ 0,75	1
> 0,75	0

7.8.8 Unused error correction

The Unused Error Correction parameter tests the extent to which regional or spot damage in the symbol has eroded the reading safety margin that error correction provides.

Decode the binarised image using the reference decode algorithm.

The amount of Unused Error Correction is calculated as $UEC = 1{,}0 - ((e + 2t) / E_{cap})$, where e = the number of erasures, t = the number of errors and E_{cap} = the error correction capacity of the symbol (the number of error correction codewords minus the number of error correction codewords reserved for error detection). If no error correction has been applied to the symbol, and if the symbol decodes, the value of *UEC* is taken as 1. If $(e + 2t)$ is greater than E_{cap}, $UEC = 0$. In symbols with more than one (e.g. interleaved) error correction block, *UEC* shall be calculated for each block independently and the lowest value shall be used for grading purposes.

Unused Error Correction shall be graded as shown in Table 10.

Table 10 — Unused Error Correction grading

UEC	Grade
≥ 0,62	4
≥ 0,50	3
≥ 0,37	2
≥ 0,25	1
< 0,25	0

7.8.9 Additional grading parameters

Symbology or application specifications may define additional parameters which may be graded and taken into account in the calculation of the overall symbol grade.

NOTE For example, an application specification may require that the *X* dimension is within a certain range.

7.9 Scan grading

The scan grade for each scan shall be the lowest grade of any parameter in that scan as measured in accordance with 7.8.2 to 7.8.9.

In order to determine the causes of poor quality grades, it is necessary to examine the grades for each parameter in the scan in question as described in Annex C.

Table 11 summarises the test parameters and grade levels.

Table 11 — Test parameters and values

Para-meter Grade	Decode	Symbol Contrast	Fixed Pattern Damage	Axial Non-uniformity	Grid Non-uniformity	Modulation and Reflectance Margin (interim values)	Unused Error Correction
4 (A)	Passes	$SC \geq 0{,}70$	See symbology specification or Annex A for grade thresholds	$AN \leq 0{,}06$	$GN \leq 0{,}38$	See 7.8.4	$UEC \geq 0{,}62$
3 (B)		$SC \geq 0{,}55$		$AN \leq 0{,}08$	$GN \leq 0{,}50$		$UEC \geq 0{,}50$
2 (C)		$SC \geq 0{,}40$		$AN \leq 0{,}10$	$GN \leq 0{,}63$		$UEC \geq 0{,}37$
1 (D)		$SC \geq 0{,}20$		$AN \leq 0{,}12$	$GN \leq 0{,}75$		$UEC \geq 0{,}25$
0 (F)	Fails	$SC < 0{,}20$		$AN > 0{,}12$	$GN > 0{,}75$		$UEC < 0{,}25$

7.10 Overall Symbol Grade

If incorrect data is obtained, then the overall symbol grade, irrespective of the other parameter grades, shall be 0. Otherwise, the overall symbol grade shall be the lowest of the individual parameter grades. Overall symbol grades shall be expressed on a numeric scale ranging in descending order of quality from 4,0 to 0,0.

NOTE The overall grade may be expressed as a real number to one decimal place in keeping with historical precedent.

7.11 Print growth

Print Growth tests that the graphical features comprising the symbol have not grown or shrunk from nominal so much as to hinder readability with less favourable imaging conditions than the test condition. The print growth parameter, the extent to which dark or light markings appropriately fill their module boundaries, is an important indication of process quality which affects reading performance. Print growth may be measured and evaluated independently in more than one axis, to determine, for example, both horizontal and vertical growth. Print growth shall not be a graded parameter but should be reported as an informative measure for the purposes of process control.

Starting with the binarised image, identify the graphical structures particular to the symbology that are most indicative of element growth or shrinkage in each axis of the symbol, which will generally be fixed structures or isolated elements. Based on the symbology specification and its reference decode algorithm, determine for each of these structures, in each axis, its nominal dimension D_{NOM} in modules.

Determine the actual dimension D in terms of X between the two edges of the structure by counting pixels along the grid lines derived by the use of the reference decode algorithm and passing through each structure to be measured in the symbol axis in question.

In each scan of the symbol, print growth shall then be calculated for each axis as the arithmetic mean of all values of (D - D_{NOM}). It shall be reported as the arithmetic mean of the values of print growth for each scan. Where the result is negative, it represents print loss.

8 Measurement methodologies for composite symbologies

Each component shall be measured and graded separately. The linear component shall be measured and graded in accordance with ISO/IEC 15416. When the two-dimensional composite component uses a multi-row bar code symbology, then the methodology specified in Clause 6 shall be applied to the two-dimensional composite component; when it uses a two-dimensional matrix symbology, then the methodology specified in Clause 7 shall be applied to it. Both the overall grade for the linear component so measured and the overall

grade for the two-dimensional composite component shall be reported, to assist users who may only require to read the linear component as well as those requiring to read the complete symbol.

9 Substrate characteristics

Certain characteristics of the substrate, notably gloss, low opacity and the presence of an over-laminate in the case of symbols printed on paper or similar media, and the surface texture and its response to the marking methods used, in the case of symbols directly marked on to the surface of an item, may affect reflectance measurements, and the recommendations in Annex E should be taken into account if any of these factors is present.

Annex A
(normative)

Symbology-specific parameters and values for symbol grading

A.1 Application

Because of differences in symbology structures and reference decode algorithms, the specific grading rules to apply to each symbology (especially with respect to fixed pattern damage) must be defined and specified for each particular symbology, either in this International Standard or within the Symbology Specification for that particular symbology.

This annex defines values corresponding to grade thresholds for Fixed Pattern Damage for Maxicode (ISO/IEC 16023). The first edition publication of this international standard also defined the fixed pattern damage grading parameters for Data Matrix and QR Code but these definitions are now included in the symbology specifications.

Where a symbology specification specifies the basis for grading these parameters, and makes express reference to this International Standard, the basis or values in the symbology specification shall override those indicated in this Annex.

Some symbologies may require additional parameters. These shall be added to the quality assessment of this standard in accordance with 7.8.9.

A.2 Data Matrix Fixed Pattern Damage

Data Matrix Fixed Pattern Damage (FPD) shall be assessed in accordance with ISO/IEC 16022.

NOTE The original version of this International Standard contained details of fixed pattern grading for Data Matrix. Such details are now found in ISO/IEC 16022.

A.3 Maxicode Fixed Pattern Damage

A.3.1 Features to be assessed

The Fixed Patterns of a Maxicode symbol are (a) a 3-ring circular bullseye near the centre of the symbol and (b) six 3-module orientation patterns surrounding it. These are shown in Figure A.1.

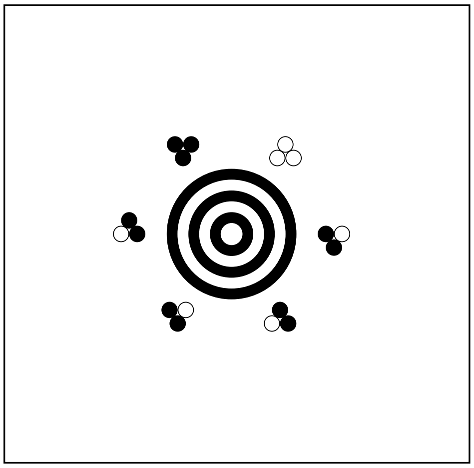


Figure A.1 — Fixed Patterns within a Maxicode symbol

A.3.2 Grading of bullseye

The bullseye is not a natural extension of the hexagonal array of data modules, and thus cannot be graded by sampling module centres. Instead, two other quality measures are performed.

1. Ring Continuity. Each of the three dark rings in the bullseye, and the two intervening light rings, shall be sampled at every image pixel location along a circular path nominally centred within the region, as shown by dotted lines in Figure A.2 below. The central light circular region shall also be sampled along a small circular path whose radius is one third the nominal radius of that region, as also shown.

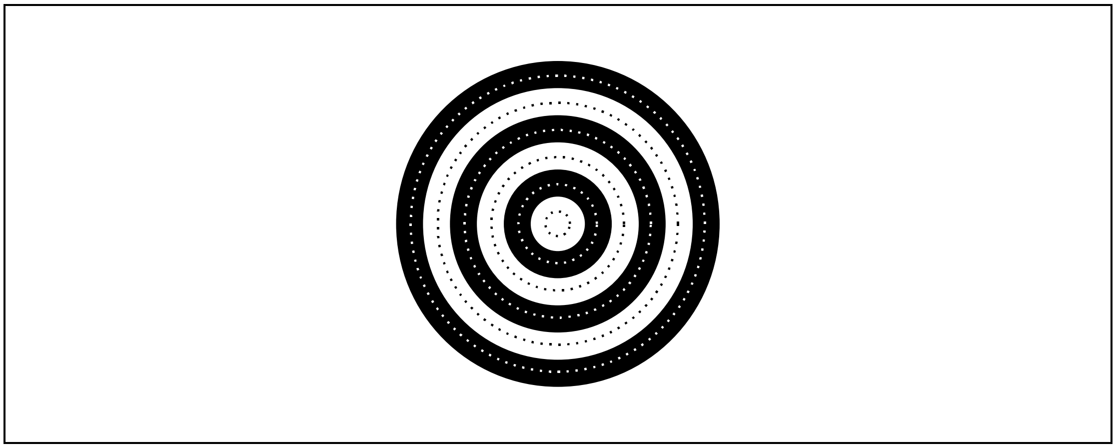


Figure A.2 — Sampling Paths within a Maxicode bullseye

These six groups of sampled points are each graded according to how many samples from along each path are the wrong colour, as a percentage of the total number of samples along that circular path, with the grade assigned as follows:

Table A.1 — Grading of Ring Continuity

Number of samples in error	Grade
0	4
≤ 3%	3
≤ 6%	2
≤ 9%	1
> 9%	0

2. <u>Ring Growth</u>. Scan profiles shall be measured from the grey-scale image along both horizontal and vertical scan paths (relative to the symbol's orientation) through the bullseye's exact centre as shown below, and the edge positions established by the methods in ISO 15416.

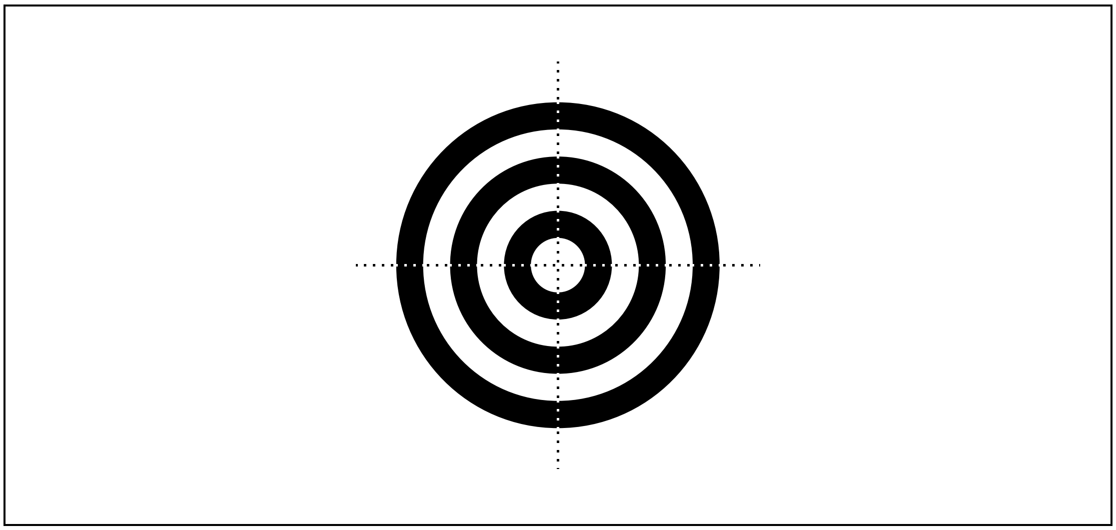


Figure A.3 — Ring Growth Sampling Paths within a Maxicode bullseye

For each profile independently, the ring growth shall be calculated as:

$$RG = (S_{bar} - S_{space}) / (S_{bar} + S_{space})$$

where S_{bar} is the sum of the bar widths

S_{space} is the sum of the space widths

excluding both of the outermost bars (the outer dark ring) and the central space (circle). These horizontal & vertical Ring Growth measurements are then each graded as:

Table A.2 — Grading of Ring Growth

RG	Grade
-0,10 < *RG* < +0,10	4
-0,14 < *RG* < +0,14	3
-0,17 < *RG* < +0,17	2
-0,20 < *RG* < +0,20	1
RG < -0,20 or *RG* > +0,20	0

A.3.3 Grading of Orientation Patterns

The six orientation patterns are taken collectively as a group of 18 modules sampled as part of the data field. Grading is performed based on a count of the number of erroneous (wrong colour) modules as follows:

Table A.3 — Grading of Orientation Patterns

Number of module errors	Grade
0	4
1	3
2	2
3	1
≥ 4	0

A.3.4 Overall Fixed Pattern Damage grade

The overall Fixed Pattern Damage grade is the lowest of the six Ring Continuity grades, the two Ring Growth grades, and the single Orientation Pattern grade achieved.

A.4 QR Code Fixed Pattern Damage and additional parameters

QR Code Fixed Pattern Damage (FPD) and additional parameters shall be assessed in accordance with ISO/IEC 18004.

NOTE The original version of this International Standard contained details of fixed pattern grading for QR Code. Such details are now found in ISO/IEC 18004.

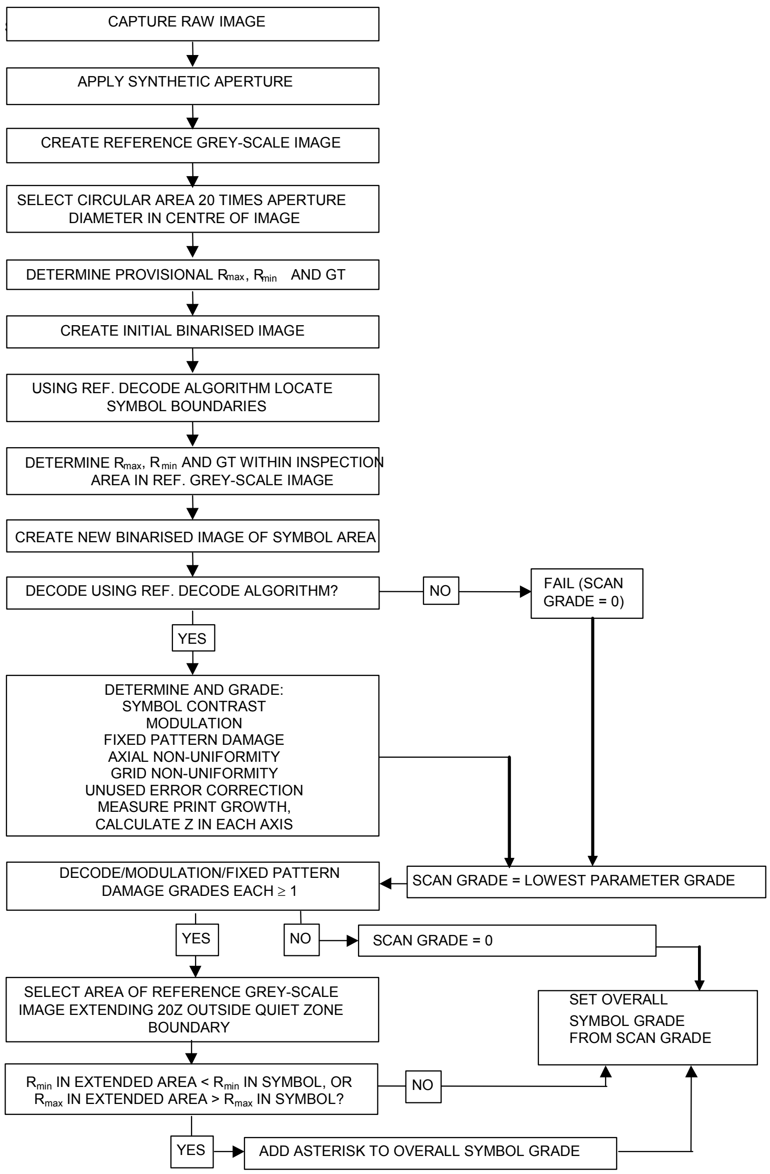
A.5 Aztec Code Fixed Pattern Damage and additional parameters

Aztec Code Fixed Pattern Damage (FPD) and additional parameters shall be assessed in accordance with ISO/IEC 24778.

Annex B
(informative)

Symbol grading flowchart for two-dimensional matrix symbols

This Annex shows the sequence of steps required in order to grade the quality of a two-dimensional matrix symbol.



JA3068

Annex C
(informative)

Interpreting the scan and symbol grades

This Annex describes possible causes of reduced grades, either in a multi-row symbol or a matrix symbol.

The table below identifies a number of factors that may lead to low or failing grades for the parameters indicated, which may be similar or differ for the two classes of two-dimensional symbol.

Table C.1 — Possible causes of low grades

Parameter	Multi-row symbols	Matrix symbols

Symbol Contrast	• low background or light module reflectance, due to:
	  • incorrect substrate e.g. blue paper for red light
	  • glossy laminate/overwrap
	  • inappropriate angle of illumination (direct marked symbols)
	• high dark module reflectance, due to
	  • low absorption of incident light by ink (unsuitable formulation/colour)
	  • insufficient ink coverage (e.g. non-overlapping ink-jet dots)
	  • inappropriate angle of illumination (direct marked symbols)
		• low background or light module reflectance, due to:
		  • incorrect substrate e.g. blue paper for red light
		  • glossy laminate/overwrap
		  • inappropriate angle of illumination (direct marked symbols)
		• high dark module reflectance, due to
		  • low absorption of incident light by ink (unsuitable formulation/colour)
		  • insufficient ink coverage (e.g. non-overlapping ink-jet dots)
		• inappropriate angle of illumination (direct marked symbols)

Decode	• many factors - see other parameters in table • software errors in printing system	• many factors - see other parameters in table • software errors in printing system
Unused Error Correction	• physical damage (scuffing, tearing, obliteration) • bit errors due to defects • excessive print growth in one or two axes • local deformation • misplaced modules	• physical damage (scuffing, tearing, obliteration) • bit errors due to defects • excessive print growth in one or two axes • local deformation • misplaced modules
Minimum Reflectance (R_{min})	• Reflectance of all bars > 0,5R_{max} - see symbol contrast for possible causes	
Minimum Edge Contrast	• excessive print growth/loss • too large measuring aperture • irregular substrate reflectance • low ink coverage • showthrough	
Modulation	• print growth/loss • too large measuring aperture • irregular substrate reflectance • variation in ink coverage • showthrough	• print growth or loss • too large measuring aperture • misplaced modules • defects (spots or voids) • irregular substrate reflectance • variation in ink coverage • showthrough
Defects	• spots of ink or other dark marks on background • voids in printed areas • faulty print head elements • too small measuring aperture	

Parameter	Multi-row symbols	Matrix symbols
Decodability	• local distortion • pixel errors in printing • slippage during printing • blocked inkjet nozzle • faulty thermal element	
Codeword Yield	• excessive tilt of scan line • *Y* axis print growth • thermal "drag"	
Fixed Pattern Damage		• blocked printer nozzle • faulty thermal element • physical damage (tearing, scuffing, obliteration)
Axial Nonuniformity		• mismatch of transport speed in printing with symbol dimensions • printing software errors • verifier axis not perpendicular to symbol plane
Grid Nonuniformity		• transport errors in printing (acceleration/deceleration, vibration, slippage) • variation in printhead to substrate distance • verifier axis not perpendicular to symbol plane
Print Growth/Loss (ungraded)	• print process-dependent factors • absorbency of substrate • dot size (ink-jet, dot peening etc.) • incorrect thermal print head temperature	• print process-dependent factors • absorbency of substrate • dot size (ink-jet, dot peening etc.) • incorrect thermal print head temperature

Annex D
(informative)

Guidance on selection of grading parameters in application specifications

D.1 Selection of measurement wavelength

D.1.1 General considerations

Clauses 6 and 7 of this standard require measurements to be made using light of the same characteristics as those which the intended scanning environment will use. If, as may happen, an application specification does not specify the light source, a judgment must be made in order to determine the most probable light source for reading, to enable valid measurements to be made and to be sure that the results will be properly indicative of likely scanning performance in the application.

It should be noted that for maximum correlation, it is not only the light source (including any filters that modify its spectral distribution) that must be taken into account, but also the spectral sensitivity of the sensor, since reflectance at a given wavelength is a function of the product of the intensity of the light emitted and the sensitivity of the sensor. However for the purposes of this Annex, the sensor sensitivity is ignored.

D.1.2 Light sources

Light sources for bar code scanning applications normally fall into two areas:

— narrow band illumination in either the visible or the infra-red spectrum or

— broadband illumination covering a large part of the visible spectrum, sometimes referred to as "white light" although it may have a bias to a colour; a very few specialised applications may call for light sources of unusual characteristics such as ultra-violet for fluorescent symbols.

Multi-row bar code scanning almost always uses narrow band visible light, with light sources with a peak wavelength in the red part of the spectrum, between 620 and 700 nm. Infra-red scanning uses sources with peak wavelengths between 720 nm and 940 nm.

Two-dimensional matrix symbols are scanned under a variety of illumination conditions, with the most common being white light and, in a number of hand-held reading devices, the same visible red area of the spectrum as for linear and multi-row bar code symbols.

The most common light sources used for these purposes are:

a) Narrow band

   1) Helium-neon laser (633 nm) (multi-row bar code symbols only)

   2) Light-emitting diode (near-monochromatic, at numerous visible and infra-red peak wavelengths)

   3) Solid-state laser diode (most usually 660 nm and 670 nm) (multi-row bar code symbols only)

b) Broadband

   1) Incandescent lamp (nominally white light with a colour temperature in the range 2800°K to 3200°K)

   2) Fluorescent lighting (nominally white light with a colour temperature in the range of 3200°K to 5500°K)

3) Light-emitting diode (nominally white light with a colour temperature in the region of 7000°K)

4) Halogen lamps (nominally white light with a colour temperature in the region of 2800°K to 3200°K)

5) Gas discharge lamps (light of various characteristics)

The key characteristics of these are as follows.

A **helium-neon laser** is a gas-filled laser tube which emits highly monochromatic coherent light at a peak wavelength of 632,8 nm (usually rounded to 633 nm), in the visible red area of the spectrum.

A **light-emitting diode** is a low-power solid-state component most frequently found as the light source in a light pen (wand) or CCD scanner. Operating wavelengths in the visible spectrum may be from 620 to 680 nm; most commonly either 633/640 or about 660 nm. In the infra-red spectrum, 880 to 940 nm is the most common range of wavelengths.

A **laser diode** is also a low power solid-state component emitting highly monochromatic coherent light. Typical wavelengths in the visible spectrum used by these, at the date of publication of this standard, are 660 and 670 nm. In the infra-red spectrum 780 nm is common. They are frequently found in hand-held (laser) scanning equipment and a number of fixed scanners.

Broadband light sources are mainly found in systems using two-dimensional imaging and image processing technology rather than scanning techniques.

Incandescent lamps have a power distribution covering much of the visible spectrum and well into the near infra-red spectrum; their optical characteristics are more easily defined in colour temperature terms rather than in those of peak wavelength, because of the wide bandwidth and relative absence of clearly-defined peaks in the power distribution. These broadband power distribution characteristics mean that the symbol contrast values obtained from symbols may vary with different colour temperatures to a significantly lesser extent than values obtained with light sources whose power distributions peak sharply with narrow bandwidth.

Halogen lamps (also known, more correctly, as tungsten halogen lamps) are a development of incandescent lamps with a higher colour temperature and a smooth power distribution curve across the spectrum, extending well into the near infra-red.

Fluorescent light sources also produce nominally white light and have broadband power distribution characteristics, which, in comparison with those of an incandescent source, tend more towards the bluer region of the visible spectrum, often with a significant ultra-violet component, and a number of peaks in their spectral power distribution. Typical colour temperatures for such lighting are in the region of 3200° to 5500°K. The physical structure of a fluorescent lamp is that of a tube which can be formed into various shapes, and an annular shape concentric with the optical axis of a reading device provides very satisfactory uniform diffuse illumination.

Light emitting diodes with nominally "white light" characteristics emit "cool" white light and may have a nominal colour temperature in the region of 7000°K. Their actual spectral distribution may show a number of peaks e.g. in the blue and yellow or orange regions.

Gas discharge lamps tend to have spectral distributions with multiple sharp peaks at wavelengths depending on the precise mixture of gases used. For example, sodium vapour emits light with a well-defined peak at around 580 nm (yellow-orange) and mercury vapour emits a green-blue light at around 520 nm.

The use of filters to modify the spectral distribution of the illumination system is common. For example, when used in conjunction with a Wratten 26 filter, the light characteristics of a 2856°K lamp approximate to those of a 620-633 nm source. The use of infra-red and/or ultra-violet absorbing filters is also common in scanning systems. It is possible to alter the apparent colour temperature of a source by the use of filters.

NOTE: Wavelengths and colour temperatures stated above are indicative and may change as the technology evolves.

D.1.3 Effect of variations in wavelength

The reflectance of a substrate or bar code symbol element varies with the wavelength of the incident light. A black, blue, or green printed area will tend to absorb visible red light strongly (and appear therefore of low reflectance), whereas a white, red or orange area will reflect most of such incident light. In the infra-red spectrum, the apparent colour of the element does not correlate at all with reflectance; it is the nature of the pigmentation used (for example the proportion of carbon content) which governs reflectance. Taking reflectance measured at 633 nm as a reference, when measured at 660 or 680 nm the results may differ significantly, and sufficiently to cause the symbol grade to change by one or two units, or even more in the case of bars printed on some thermal papers.

In the case of broadband illumination, however, the presence of light at multiple wavelengths in the spectral power distribution of the light means that reflectance values of black inks measured under white light from various sources will not differ significantly; there may however be some variation (an increase in reflectance) in the case of dye-based black inks where the illumination has a significant infra-red component. With coloured pigments there will be greater variations. Interposing a filter in the light path will introduce a more peaked spectral distribution and the spectral response curve of the reader will require to be more closely matched to that of the light source. It is not uncommon for the optical train to include both infra-red and ultra-violet absorbing filters.

D.1.4 Considerations affecting selection of broadband light sources

Broadband light sources, by definition, emit light over a band of wavelengths without a clearly defined sharp peak. Nonetheless, the intensity of light emitted at different wavelengths will vary. In particular, light of a colour temperature in the region of 3000°K is described as "warm" light and the spectral distribution of this light shows higher intensity of emission towards the red (and infra-red) region of the spectrum, whereas light with a higher colour temperature in the region of 6500°K is described as "cool" light and its spectral distribution is biased to the blue-violet region of the spectrum, extending into the ultra-violet. Light with a higher colour temperature will yield higher reflectance values on blue pigments than light with a lower colour temperature. The converse is true for red pigments.

It is possible to modify the apparent colour temperature of a light source by the interposition of an appropriate filter.

It may also be possible to approximate the characteristics of different broadband light sources with sufficient precision for bar code symbol quality assessment purposes by combining reflectance measurements at three narrow band wavelengths across the visible spectrum, e.g. in the red, green and blue regions (assuming that the ultra-violet and infra-red regions have been cut off by the use of appropriate absorbing filters); the results can then be modified to match the spectral response characteristics in the application by applying an appropriate correction factor at each wavelength.

D.2 Selection of aperture

For matrix symbol grading, the choice of aperture size is very important, and it must be specified in accordance with 7.3.3 in order for symbol grades to be measured consistently. It is the responsibility of an application specification to define a fixed measuring aperture to be used. As required by 5.4 of this standard, the aperture size must be reported together with the grade and illumination in order to identify the conditions under which the measurement was made.

The size of the measuring aperture affects whether voids in the symbol will be “filled in” during the verification process. Therefore, the measuring aperture must be selected with reference to the range of nominal module size and expected scanning environment. An aperture that is too small will not fill in unintentional voids, or gaps between elements of a direct marked symbol, that would lead to low grades or undecodable symbols. On the other hand, a measuring aperture that is too large will blur individual modules, resulting in low modulation, and may prevent the symbol from being decoded.

An aperture size in the range of 50% to 80% of the minimum allowed module size is a typical choice for an application specification. Importantly however, an application specification that allows a range of nominal

module sizes (for example a range of 0,25 mm through 0,40 mm) should specify a single aperture size to be used in all cases. That is to say that the verification of each symbol is made with an aperture that is not necessarily related to that symbol's module size. For example if 80% of 0,25 mm were specified, i.e. a 0,20 mm aperture, then all symbols used in that application including 0.40 mm symbols must be measured using a 0,20 mm aperture. Another aperture size may be chosen, even one equal to or larger than the minimum module size. The important factor is that a single measuring aperture be specified and used consistently within an application area.

If a range of module sizes is used within an application specification, then the relatively small measuring aperture required to read the symbols with the smallest module size will limit the size of the largest acceptable spots and voids. If too large an aperture were used, the modulation for the smallest module size would be inadequate. In general the larger the aperture, the larger the acceptable size of spots and voids. Conversely, the smaller the aperture, the smaller the acceptable module size that can be read. Therefore, a successful application specification must select a measuring aperture that will predict the readability of both the largest and smallest module size symbols.

A single fixed measuring aperture ensures that all symbols will be measured in a way that will reflect performance in the expected scanning environment. The choice of measuring aperture that is specified will be influenced in some cases by the scanning equipment that is expected to be prevalent in the application-scanning environment. Conversely, the scanning equipment may also be influenced by the specification of the measuring aperture. In both cases however, a “match” between verification technique and scanning environment is made in order to produce a high correlation between grade level and scanning performance.

The nominal diameter of the measuring aperture should be specified by the user application specification, to suit the intended scanning environment or with reference to the guidelines of Table D.1 . When the measuring aperture diameter is not specified in the application specification, Table D.1 should be used as a guide. In an application where a range of X dimensions will be encountered, all measurements shall be made with the aperture appropriate to the smallest X dimension to be encountered.

NOTE An application specification may specify a range of X dimensions that differs from those listed in Table D.2 and can specify an aperture size that differs from the recommendation of Table D.2.

Table D.1 — Guideline for diameter of measuring aperture

X Dimension(mm)	Aperture diameter (mm)	Reference number
$0{,}100 \leq X < 0{,}150$ (4mil-6mil)	0,050	02
$0{,}150 \leq X < 0{,}190$ (6mil-7.5mil)	0,075	03
$0{,}190 \leq X < 0{,}250$ (7.5mil-10mil)	0,125	05
$0{,}250 \leq X < 0{,}500$ (10mil-20.0mil)	0,200	08
$0{,}500 \leq X < 0{,}750$ (20mil-30mil)	0,400	16
$0{,}750 \leq X$ (30mil-)	0,500	20
NOTE The aperture reference number approximates to the measuring aperture diameter in thousandths of an inch; this reference number is used for consistency with the ANSI standard X3.182 and ISO/IEC 15416.		

D.3 Selection of lighting angle

Whereas the default 45° illumination angle is well suited to printed symbols and those marked on even surfaces without localised points of specular reflection, i.e. those from which diffuse reflection does not vary sharply with the angle of incidence or collection of the light, many "direct marked" symbols require to have the angle of incidence adjusted to optimise reading performance. The spectral characteristics of the illumination for use with engraved or similar symbols may therefore be less important than the respective angles of incidence and collection of the light on the symbol. The light source needs to be positioned in such a way that the apparent contrast as seen by the image capture device is related to the process used in the application to read the symbols.

Depending on the nature of the surface and the marking technique used, the spectral characteristics of the light source may also have an influence on its contrast. Prediction of scanner performance will be enhanced when the application specification requires verification using the same source as that used for reading the symbols. The modified version of the methodology defined in ISO/IEC TR 29158 may provide a better approach to select appropriate lighting angles for direct part marks.

D.4 Selection of minimum acceptable grade

The specification of the minimum acceptable grade in an application specification should be based on consideration of the trade-off between the possibly increased cost of producing higher-grade symbols and the improved scanning performance to be obtained by the use of such symbols, together with the data integrity requirements of the application.

A requirement for higher grade symbols may restrict the choice of the following available to the producer of the symbols:

— Inks (or other marking media) and substrates on which to mark the symbol (e.g. to ensure a high level of symbol contrast, a substrate with high reflectance and/or an ink with low reflectance under the specified illumination are needed, imposing limits on the choice of colour available)

— Marking technology (e.g. those in which the placement of printer dots is less well-controlled may be excluded)

It may also require slower production rates or higher levels of quality control, or lead to higher rejection rates, all of which contribute to higher unit costs.

On the other hand, the receiver of the symbols will benefit from improved read rates, or may have a greater choice of reading technology open to him.

If a low symbol grade is specified, the receiver of the symbols may incur additional costs in:

— Installing higher quality reading equipment

— Accepting a lower read rate

— Reprocessing of symbols that failed to scan.

Many applications require a minimum grade of 1,5 (C), which offers them a fair balance between the cost of production and reading performance under the conditions of their application.

The more critical it is that a high read rate be achieved, on grounds not only of cost but also data integrity, the higher the grade that needs to be specified.

D.5 Affect of Symbol Rotation during verification

The first edition publication of this international standard called for an overall grade to be computed by averaging five individual scan results, taken with the symbol rotated in five different orientations. However, this requirement has been removed and this standard now calls for only one scan to be used for grading.

The five scan rotation requirement was originally required for two reasons: to account for symbols that do not exhibit uniform diffuse reflection and to “average out” any effective changes in measurement resolution by verification devices. The first reason is better addressed through a modified version of the methodology defined in this international standard in ISO/IEC TR 29158. The second reason is addressed by the publication of ISO/IEC 15426-2 which requires the minimum effective resolution defined in 7.3.3.

The removal of the five rotation requirement greatly simplifies the verification procedure for most symbols (whose characteristics do not vary with orientation) and facilitates practical quality control regimes.

Annex E
(informative)

Substrate characteristics

E.1 Rationale

In certain circumstances for example, the design and production of printed packaging materials incorporating bar code symbols or the production of symbols directly marked on to a surface, it may be necessary or desirable to assess the acceptability of substrates and/or ink colours for a given bar code application, before a bar code symbol is available which could be tested in accordance with this standard. Reference should be made to ISO/IEC TR 19782 for additional guidance on the effects of gloss and low substrate opacity on the reading and verification of these symbols.

E.2 Substrate opacity

The methodology of the International Standard requires that the symbol shall be graded according to the parameters in Clause 6 (multi-row symbols) or 7 (matrix symbols) when measured in its final configuration, e.g. final filled package.

If it is not possible to measure the symbol in this configuration then the effects of show-through of high-contrast interfering patterns may be ignored if when measured as follows the substrate opacity is 0,85 or greater. If the opacity is less than 0,85 the symbol should be measured when backed by a uniform dark surface the reflectance of which is not more than 5 per cent.

The opacity of the substrate shall be calculated as follows:

$$\text{Opacity} = R2 / R1$$

where: $R1$ = Reflectance of a sample sheet of the substrate backed with a white surface the reflectance of which is 89 percent or greater.

$R2$ = Reflectance of the same sample sheet backed with a black surface the reflectance of which is not more than 5 percent.

E.3 Gloss

The reference illumination conditions specified for the measurement of reflectance should enable the maximum rejection of specular reflection while giving a representative assessment of the diffuse reflectance of the symbol and substrate. Highly glossy materials and those whose diffuse reflectance characteristics vary with the angle of incident and/or collected light - as may be the case with many materials on which symbols are directly marked - may yield grades differing from those obtained by the use of the reference optical arrangement with illumination at 45°, and for this reason sub-clause 7.3.4 provides alternative angles of illumination to enable apparent symbol contrast to be maximised.

E.4 Over-laminate

A symbol intended to be covered with a protective lamination should be graded according to the parameters in clause 6 (multi-row symbols) or Clause 7 (matrix symbols) when measured with the laminate in place. The thickness of the laminate including its adhesive should be as small as possible in order to minimise its effects on the reading performance of the symbol.

E.5 Static reflectance measurements

E.5.1 General

In some cases it may be desirable to carry out static reflectance measurements of samples of the substrate on which a bar code is to be printed and on colour patches or ink samples which replicate the colour in which the bar code will be printed. The following guidelines provide a means which, if it is followed, will predict as closely as is generally possible the results which will be obtained when the symbol is scanned dynamically.

Static reflectance measurements should be made with the wavelength of light, aperture size and optical arrangement which relate to the application and which are specified in accordance with ISO/IEC 15416 (multi-row symbols) of this standard.

Where reflectance measurement equipment meeting the requirements of this Annex is not available, optical density measurements may be made using a standard densitometer with an appropriate light source and converted to reflectance values; density (D) and reflectance (R) are related as follows:

$$R = 100 / 10^D$$

NOTE It is impossible to predict to a high degree of accuracy the symbol contrast and, in particular, the edge contrast which will be achieved in the printed symbol. It is therefore appropriate to allow some safety margin above the minimum values for specified grades.

E.5.2 Prediction of Symbol Contrast (*SC*)

The prediction of *SC* requires that measurements of reflectance be made on samples which simulate the highest (R_{max}) and lowest reflectance (R_{min}) areas which will be present in the finished symbol.

It is probable that in most bar code symbols R_{max} will be found in the quiet zone of the symbol; therefore to simulate the conditions found in the quiet zone, R_{max} should be measured in the centre of a sample area, at least 10X in diameter, of the material on which the symbol is to be printed.

It is probable that in most bar code symbols R_{min} will be found in the widest bars of the symbol, or areas with a number of contiguous dark modules; therefore to simulate the conditions most likely to yield a value of R_{min} consistent with that which would be found in practice, reflectance should be measured in the centre of a strip of material 2X to 3X wide and which matches the colour in which the dark elements are to be printed.

A predicted value of *SC* can then be calculated:

$$SC' = R_{max} - R_{min}$$

For materials which do not satisfy the tests for opacity, which are detailed in Annex E.1, the measurements which are made for the purpose of predicting *SC* should be made with the test samples backed by a uniform dark surface, the reflectance of which is not more than 5%. The same measurements should then be made with the test samples backed by a uniform surface the reflectance of which is not less than 89%. The calculated value of static SC shall be equal to or greater than the minimum value for the grade selected for the application, for tests on both the dark and light backgrounds.

Annex F
(informative)

Parameter grade overlay applied to two-dimensional symbologies

This Annex describes the technique used in this International Standard to derive a final grade for a parameter from a set of notional grades determined for a set of grade levels, each determined at five fixed grade levels for the parameter.

The technique computes a notional grade for a parameter for each grade level by assuming that only modules or codewords that meet or exceed that grade level for that parameter are actually readable. The modules or codewords which are readable then result in a grade for that parameter according to the rules for that parameter (whether based on unused error correction or fixed pattern damage).

If one considers what the performance would be for a scanner that could only read codewords or modules above a particular parameter grade level, it is clear what will happen – only codewords or modules at or above that grade level may be counted towards the readability calculation for the symbol at that grade level.

For example, if codewords or modules with a grade of 2 must be counted before a grade of 3 on unused error correction or fixed pattern damage may be obtained, then the symbol must be a grade 2.

Furthermore, if codewords graded 3 or better can only result in an unused error correction or fixed pattern damage level of 2, the symbol must also be a grade 2.

However, the readability of a symbol must take into account the readability of codewords or modules at each grade level and the ability of the symbol to be read using error correction or allowing for some fixed pattern damage and the resulting grade should be the highest of these two possible outcomes.

The following procedure can therefore be established:

a) Count the number of codewords in each grade level, including higher grade levels, assume that all remaining codewords are erasures (multi-row symbols) or errors (matrix symbols) and determine an unused error correction or fixed pattern damage grade.

b) For each grade level, take the lower of the grade level and the associated unused error correction or fixed pattern damage parameter grade.

c) Select the highest of the values from step b as the symbol grade for that parameter.

This ensures that a scanner will have performance associated with the assigned grade because the scanner's ability to read codewords or modules of the assigned grade or higher will bring it within the error correction or fixed pattern damage capacity of the assigned grade level or better.

This method provides a way of accounting for imperfections in symbols which are designed to tolerate imperfections. In fact, it favours symbols with more error correction capacity, which certainly does make a symbol easier to read. It also reconciles the print quality measurement method of linear symbols with that of 2D symbols. In a sense the linear approach, which takes the worst case, is the trivial extension of the above rule in the case of no error correction. In this case, the codeword with lowest grade is always needed to get anything other than a 0 for "unused error correction". If this value happens to be a 1, then the symbol must be a 1, even if all other codewords had quality of 4.

NOTE The notional Unused Error Correction or Fixed Pattern Damage grade used in this calculation is not related to, and does not affect, the UEC or Fixed Pattern Damage grade for the symbol as calculated according to 6.2.4, 7.8.8 or 7.8.5 respectively.

Bibliography

Symbology specifications

Multi-Row Symbols with cross-row scanning ability

[1] ISO/IEC 15438, *Information technology — Automatic identification and data capture techniques — PDF417 bar code symbology specification*

[2] ISO/IEC 24728, *Information technology — Automatic identification and data capture techniques — MicroPDF417 bar code symbology specification*

[3] AIM ITS - SuperCode

Multi-Row Symbols without cross-row scanning ability

[4] EN 12323, AID technologies - Symbology Specifications - Code 16K

[5] ANSI/AIM BC6-1995, USS - Code 49

[6] AIM USS Codablock F

[7] ISO/IEC 24724, *Information technology — Automatic identification and data capture techniques — GS1 DataBar bar code symbology specification*

Matrix Symbols

[8] ISO/IEC 16022, *Information technology — Automatic identification and data capture techniques — Data Matrix bar code symbology specification*

[9] ISO/IEC 16023, *Information technology — International symbology specification — MaxiCode*

[10] ISO/IEC 18004, *Information technology — Automatic identification and data capture techniques — QR Code 2005 bar code symbology specification*

[11] AIM USS - Code One

[12] ISO/IEC 24778, *Information technology — Automatic identification and data capture techniques — Aztec Code bar code symbology specification*

[13] AIM USS - Dot Code A

Composite Symbologies

[14] ISO/IEC 24723, *Information technology — Automatic identification and data capture techniques — GS1 Composite bar code symbology specification*

[15] AIM ITS - Aztec Mesas

NOTE 1 Specifications published by AIM Global - viz. AIM International Technical Specifications (ITS) and Uniform Symbology Specifications (USS) - are obtainable from AIM Inc., 634 Alpha Drive, Pittsburgh, PA 15238, USA

NOTE 2 This is not an exhaustive list of symbology specifications.

Other specifications

[16] ISO 2859-10, *Sampling procedures for inspection by attributes — Part 10: Introduction to the ISO 2859 series of standards for sampling for inspection by attributes*

[17] ISO 3951-1, *Sampling procedures for inspection by variables — Part 1: Specification for single sampling plans indexed by acceptance quality limit (AQL) for lot-by-lot inspection for a single quality characteristic and a single AQL*

[18] ISO 3951-2, *Sampling procedures for inspection by variables — Part 2: General specification for single sampling plans indexed by acceptance quality limit (AQL) for lot-by-lot inspection of independent quality characteristics*

[19] ISO 3951-3, *Sampling procedures for inspection by variables — Part 3: Double sampling schemes indexed by acceptance quality limit (AQL) for lot-by-lot inspection*

[20] ISO 3951-5, *Sampling procedures for inspection by variables — Part 5: Sequential sampling plans indexed by acceptance quality limit (AQL) for inspection by variables (known standard deviation)*

[21] ISO/IEC 15426-1, *Information technology — Automatic identification and data capture techniques — Bar code verifier conformance specification — Part 1: Linear symbols*

[22] ISO/IEC 15426-2, *Information technology — Automatic identification and data capture techniques — Bar code verifier conformance specification — Part 2: Two-dimensional symbols*

[23] ISO/IEC TR 19782, *Information technology — Automatic identification and data capture techniques — Effects of gloss and low substrate opacity on reading of bar code symbols*

[24] ISO/IEC TR 29158, *Information technology — Automatic identification and data capture techniques — Direct Part Mark (DPM) Quality Guideline*

ICS 35.040
Price based on 43 pages

INTERNATIONAL STANDARD

ISO/IEC 15438

Third edition
2015-09-15

Information technology — Automatic identification and data capture techniques — PDF417 bar code symbology specification

Technologies de l'information — Techniques automatiques d'identification et de capture des données — Spécifications pour la symbologie de code à barres PDF417

EXHIBIT
1005

Reference number
ISO/IEC 15438:2015(E)

JA3083

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: Foreword - Supplementary information

The committee responsible for this document is ISO/IEC JTC 1, *Information technology*, Subcommittee SC 31, *Automatic identification and data capture techniques*.

This third edition cancels and replaces the second edition (ISO/IEC 15438:2006), of which it constitutes a minor revision.

Introduction

The technology of bar coding is based on the recognition of patterns of bars and spaces of defined dimensions. There are various methods of encoding information in bar code form, known as symbologies, and the rules defining the translation of characters into bars and space patterns and other essential features are known as the symbology specification.

Manufacturers of bar code equipment and users of bar code technology require publicly available standard symbology specifications to which they can refer when developing equipment and application standards. It is the intent and understanding of ISO/IEC that the symbology presented in this International Standard is entirely in the public domain and free of all user restrictions, licences and fees.

Information technology — Automatic identification and data capture techniques — PDF417 bar code symbology specification

1 Scope

This International Standard specifies the requirements for the bar code symbology known as PDF417. It specifies PDF417 symbology characteristics, data character encodation, symbol formats, dimensions, error correction rules, reference decoding algorithm, and a number of application parameters.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 646, *Information technology — ISO 7-bit coded character set for information interchange*

ISO/IEC 15415, *Information technology — Automatic identification and data capture techniques — Bar code symbol print quality test specification — Two-dimensional symbols*

ISO/IEC 15424, *Information technology — Automatic identification and data capture techniques — Data Carrier Identifiers (including Symbology Identifiers)*

ISO/IEC 19762-1, *Information technology — Automatic identification and data capture (AIDC) techniques — Harmonized vocabulary — Part 1: General terms relating to AIDC*

ISO/IEC 19762-2, *Information technology — Automatic identification and data capture (AIDC) techniques — Harmonized vocabulary — Part 2: Optically readable media (ORM)*

ISO/IEC 24723, *Information technology — Automatic identification and data capture techniques — GS1 Composite bar code symbology specification*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 19762-1, ISO/IEC 19762-2 and the following apply.

3.1
basic channel model
standard system for encoding and transmitting bar code data where data message bytes are output from the decoder but no control information about the message is transmitted

Note 1 to entry: A decoder complying with this model operates in Basic Channel Mode.

3.2
bar-space sequence
sequence which represents the module widths of the elements of a symbol character

3.3
cluster
any of the three mutually exclusive subsets of PDF417 symbol characters

Note 1 to entry: The symbol characters in a given cluster conform with particular structural rules which are used in decoding the symbology.

3.4
compaction mode
any of the three data compaction algorithms in PDF417 (Text, Numeric and Byte Compaction modes) which are used to map 8-bit data bytes efficiently to PDF417 codewords

3.5
e-distance
distance from the leading edge of an element to the leading edge of the next similar element, or from trailing edge to trailing edge

3.6
error correction codeword
encodes a value derived from the error correction codeword algorithm to enable decode errors to be detected and, depending on the error correction level, to be corrected

3.7
Extended Channel Interpretation
ECI
procedure within some symbologies, including PDF417, to replace the default interpretation with another interpretation in a reliable manner

Note 1 to entry: The interpretation intended prior to producing the symbol can be retrieved after decoding the scanned symbol to recreate the data message in its original format.

3.8
Extended Channel Model
system for encoding and transmitting both data message bytes and control information about the message, the control information being communicated using Extended Channel Interpretation (ECI) escape sequences

Note 1 to entry: A decoder complying with this model operates in Extended Channel Mode.

3.9
function codeword
initiates a particular operation within a symbology

EXAMPLE To switch between data encoding sets, to invoke a compaction scheme, to program the reader, or to invoke Extended Channel Interpretations.

3.10
Global Label Identifier
GLI
procedure in the PDF417 symbology which behaves in a similar manner to Extended Channel Interpretation

Note 1 to entry: The GLI system was the PDF417-specific precursor to the symbology-independent ECI system.

3.11
Macro PDF417
procedure in the PDF417 symbology logically to distribute data from a computer file across a number of related PDF417 symbols

Note 1 to entry: The procedure considerably extends the data capacity beyond that of a single symbol.

Note 2 to entry: This procedure is similar to the Structured Append feature in other symbologies.

3.12
Mode Latch codeword
used to switch from one mode to another mode, which stays in effect until another latch or shift codeword is implicitly or explicitly brought into use, or until the end of the symbol is reached

3.13
Mode Shift codeword
used to switch from one mode to another for one codeword, after which encoding returns to the original mode

3.14
Row Indicator codeword
PDF417 codeword adjacent to the start or stop character in a row, which encodes information about the structure of the PDF417 symbol in terms of the row identification, total number of rows and columns, and the error correction level

3.15
Symbol Length Descriptor
first codeword in a PDF417 symbol, which encodes the total number of data codewords in the symbol

4 Symbols, operations and abbreviated terms

4.1 Symbols

For the purposes of this International Standard, the following mathematical symbols apply. There are some cases where the symbols below have been used in a different manner in an equation. This has been done for consistency with a more general use of the notation and is always clearly defined in the text.

A symbol aspect ratio (height to width) of a PDF417 symbol

b element width in a symbol character

c number of columns in the symbol in the data region (excluding start, stop and row indicator codewords)

d data codeword including all function codewords

E error correction codeword

e edge to similar edge dimension in a symbol character

F row number

f number of substitution errors

H height of symbol including quiet zone

K cluster number

k number of error correction codewords

L left row indicator

l number of erasures

m number of source data codewords prior to the addition of the Symbol Length Descriptor and any pad codewords

n total number of data codewords including Symbol Length Descriptor and any pad codewords

p pitch or width of a symbol character

Q_H horizontal quiet zone

Q_V vertical quiet zone

R right row indicator

r number of rows in the symbol

s error correction level

W width of symbol including quiet zone

X X-dimension or module width

Y module height (also called row height)

4.2 Mathematical operations

For the purposes of this International Standard, the following mathematical operations apply.

div is the integer division operator, rounding down

INT is the integer value, i.e. where a number is rounded down to its whole number component, ignoring its decimal fractions

mod is the positive integer remainder after division. If the remainder is negative, add the value of the divisor to make it positive. For example, the remainder of –29 160 divided by 929 is –361 which when added to 929 yields 568.

4.3 Abbreviated terms

For the purposes of this International Standard, the following abbreviated terms apply.

ECI Extended Channel Interpretation

GLI Global Label Identifier

5 Requirements

5.1 Symbology characteristics

5.1.1 Basic characteristics

PDF417 is a bar code symbology with the following basic characteristics.

a) Encodable character set:

1) Text Compaction mode (see 5.4.1.5) permits all printable ASCII characters to be encoded, i.e. values 32 to 126 inclusive in accordance with ISO/IEC 646 (IRV), as well as selected control characters;

2) Byte Compaction mode (see 5.4.3) permits all 256 possible 8-bit byte values to be encoded. This includes all ASCII characters value 0 to 127 inclusive and provides for international character set support;

3) Numeric Compaction mode (see 5.4.4) permits efficient encoding of numeric data strings;

4) Up to 811 800 different character sets or data interpretations;

5) Various function codewords for control purposes.

b) Symbol character structure: (*n, k, m*) characters of 17 modules (*n*), 4 bar and 4 space elements (*k*), with the largest element 6 modules wide (*m*).

c) Maximum possible number of data characters per symbol (at error correction level 0): 925 data codewords which can encode:

1) Text Compaction mode: 1 850 characters (at 2 data characters per codeword);

2) Byte Compaction mode: 1 108 characters (at 1,2 data characters per codeword);

3) Numeric Compaction mode: 2 710 characters (at 2,93 data characters per codeword).

At the minimum recommended error correction level, there is a maximum of 863 data codewords which can encode:

4) Text Compaction mode: 1 726 characters (at 2 data characters per codeword);

5) Byte Compaction mode: 1 033 characters (at 1,2 data characters per codeword);

6) Numeric Compaction mode: 2 528 characters (at 2,93 data characters per codeword).

d) Symbol size:

1) Number of rows: 3 to 90;

2) Number of columns: 1 to 30;

3) Width in modules: 90X to 583X including quiet zones;

4) Maximum codeword capacity: 928 codewords;

5) Maximum data codeword capacity: 925 codewords.

Since the number of rows and the number of columns are selectable, the aspect ratio of a PDF417 symbol may be varied when printing to suit the spatial requirements of the application.

e) Selectable error correction: 2 to 512 codewords per symbol (see 5.7).

f) Non-data overhead:

1) Per row: 73 modules, including quiet zones;

2) Per symbol: a minimum of 3 codewords, represented as symbol characters.

g) Code type: continuous, multi-row two-dimensional.

h) Character self-checking: Yes.

i) Bi-directionally decodable: Yes.

5.1.2 Summary of additional features

Additional features which are inherent or optional in PDF417 are summarised below.

a) **Data compaction**: (inherent) Three schemes are defined to compact a number of data characters into codewords. Generally data is not directly represented on a one character for one codeword basis (see 5.4.1.5 to 5.4.4).

b) **Extended Channel Interpretations**: (optional) These mechanisms allow up to 811 800 different data character sets or interpretations to be encoded (see 5.5).

c) **Macro PDF417**: (optional) This mechanism allows files of data to be represented logically and consecutively in a number of PDF417 symbols. Up to 99 999 different PDF417 symbols can be so linked or concatenated and be scanned in any sequence to enable the original data file to be correctly reconstructed (see 5.13).

d) **Edge to edge decodable**: (inherent) PDF417 can be decoded by measuring elements from edge to similar edge (see 5.3.1).

e) **Cross row scanning**: (inherent) The combination of the following three characteristics in PDF417 facilitates cross row scanning:

   1) being synchronised horizontally, or self clocking;

   2) row identification;

   3) being synchronised vertically, by using the cluster values to achieve local row discrimination.

   This combination allows a single linear scan to cross a number of rows and achieve a partial decode of the data so long as at least one complete symbol character per row is decoded into its codeword. The decoding algorithm can then place the individual codewords into a meaningful matrix.

f) **Error correction**: (inherent) A user may define one of 9 error correction levels. All but Level 0 not only detect errors but also can correct erroneously decoded or missing codewords (see 5.7).

g) **Compact PDF417**: (optional) In relatively 'clean' environments, it is possible to reduce some of the row overhead to improve the symbol density (see 5.12).

NOTE In earlier specifications of PDF417, Compact PDF417 was called Truncated PDF417. Compact PDF417 is the preferred term to avoid confusion with the more general use of the term 'truncated'.

5.2 Symbol structure

5.2.1 PDF417 symbol parameters

Each PDF417 symbol consists of a stack of vertically aligned rows with a minimum of 3 rows (maximum 90 rows). Each row shall include a minimum of 1 symbol character (maximum 30 symbol characters), excluding start, stop and row indicator columns. The symbol shall include a quiet zone on all four sides. Figure 1 illustrates a PDF417 symbol encoding the text: PDF417 Symbology Standard.

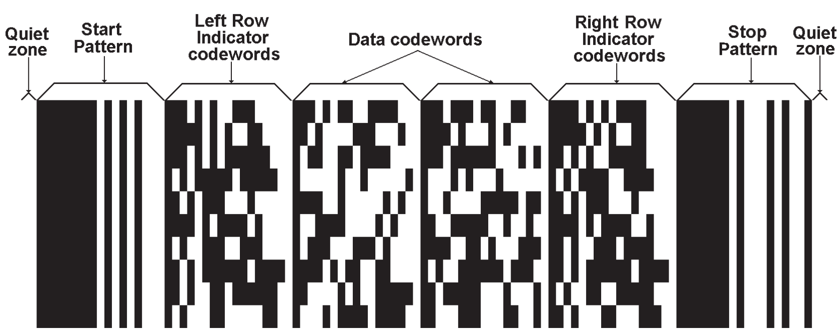


Figure 1 — PDF417 symbol structure

5.2.2 Row parameters

Each PDF417 row shall comprise of the following:

a) leading quiet zone;

b) start character;

c) left row indicator symbol character;

d) 1 to 30 symbol characters;

e) right row indicator symbol character;

f) stop character;

g) trailing quiet zone.

NOTE The number of symbol characters (or codewords) defined in item ‘d’ above is equal to the number of data columns in the PDF417 symbol.

5.2.3 Codeword sequence

A PDF417 symbol may contain up to 928 symbol characters or codewords. Symbol character is the more appropriate term to refer to the printed bar/space pattern; codeword is more appropriate for the numeric value of the symbol character. The codewords shall follow this sequence:

a) The first codeword, the Symbol Length Descriptor, shall always encode the total number of data codewords in the symbol, including the Symbol Length Descriptor itself, data codewords and pad codewords, but excluding the number of error correction codewords.

b) The data codewords shall follow, from the most significant encodable character. Function codewords may be inserted to achieve data compaction.

c) Pad codewords to enable the codeword sequence to be represented in a rectangular matrix. Pad codewords may also be used to fill additional complete rows to achieve an aspect ratio desired or as specified by the application.

d) An optional Macro PDF417 Control Block.

e) Error correction codewords for error detection and correction.

The codewords are arranged with the most significant codeword adjacent to the Symbol Length Descriptor, and are encoded from left to right and from top row to bottom. Figure 2 illustrates in layout format the sequence for a symbol like what is being shown in Figure 1. In Figure 2, an error correction level of 1 has been used and one pad character was needed to completely fill the symbol matrix.

<table>
<tr><td rowspan="10">S
T
A
R
T</td><td>L_1</td><td>d_{15}</td><td>d_{14}</td><td>R_1</td><td rowspan="10">S
T
O
P</td></tr>
<tr><td>L_2</td><td>d_{13}</td><td>d_{12}</td><td>R_2</td></tr>
<tr><td>L_3</td><td>d_{11}</td><td>d_{10}</td><td>R_3</td></tr>
<tr><td>L_4</td><td>d_9</td><td>d_8</td><td>R_4</td></tr>
<tr><td>L_5</td><td>d_7</td><td>d_6</td><td>R_5</td></tr>
<tr><td>L_6</td><td>d_5</td><td>d_4</td><td>R_6</td></tr>
<tr><td>L_7</td><td>d_3</td><td>d_2</td><td>R_7</td></tr>
<tr><td>L_8</td><td>d_1</td><td>d_0</td><td>R_8</td></tr>
<tr><td>L_9</td><td>E_3</td><td>E_2</td><td>R_9</td></tr>
<tr><td>L_{10}</td><td>E_1</td><td>E_0</td><td>R_{10}</td></tr>
</table>

Figure 2 — PDF417 Example of Symbol Layout Schematic

where

L, R, d and E	are as defined in Clause 4;
d_{15}	Symbol Length Descriptor (in this example, with a value of 16);
d_{14} to d_1	encoded representation of data;
d_0	pad codeword.

The rules and advice for structuring the matrix are included in 5.9.

5.3 Basic encodation

5.3.1 Symbol character structure

Each PDF417 symbol character shall consist of four bar elements and four space elements, each of which can be one to six modules wide. The four bar and four space elements shall measure 17 modules in total. PDF417 symbol characters can be decoded by measuring the e-distances within the character.

Each symbol character is defined by an 8-digit bar-space sequence which represents the module widths of the eight elements of that symbol character. Figure 3 illustrates a symbol character with the bar-space sequence 51111125.

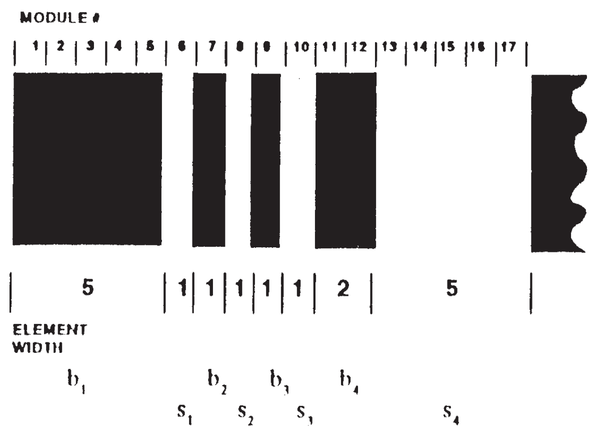


Figure 3 — A PDF417 symbol character

There are 929 defined symbol character values (codewords) numbered from 0 to 928.

The codewords are represented by three mutually exclusive symbol character sets, or clusters. Each cluster encodes the 929 available PDF417 codewords into different bar-space patterns so that one cluster is distinct from another. The cluster numbers are 0, 3, and 6. The cluster definition applies to all PDF417 symbol characters, except for start and stop characters.

The cluster number K is defined by the following formula:

$$K = (b_1 - b_2 + b_3 - b_4 +) \bmod 99$$

where b_1, b_2, b_3 and b_4 represent the width in modules of the four bar elements respectively.

The cluster number K for the symbol character in Figure 3 is:

$K = (5 - 1 + 1 - 2 + 9) \bmod 9 = 3$

The codewords and the bar-space sequences for each cluster of symbol characters are given in Annex A.

5.3.2 Start and stop characters

The start and stop characters shall be composed as defined in Table 1 and illustrated in Figure 4:

Table 1 — Bar-space sequence for Start and Stop Characters

Character	Bar-space sequence								
	B	S	B	S	B	S	B	S	B
Start	8	1	1	1	1	1	1	3	
Stop	7	1	1	3	1	1	1	2	1

NOTE 1 The PDF417 stop and start characters are unique in having elements more than 6 modules wide.

NOTE 2 The stop character has one extra single module bar element.

The start and stop characters shall have the same bar-space sequence for all rows.

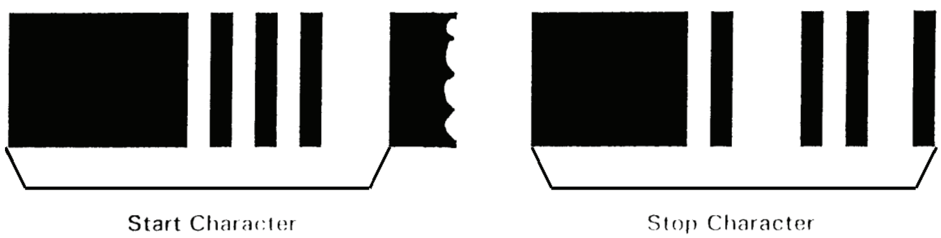


Figure 4 — PDF417 Start and Stop Characters

5.4 High level (data) encodation

High level encoding converts the data characters into their corresponding codewords.

Data compaction schemes shall be used to achieve efficient high level encoding. Three modes are defined below, each of which defines a particular efficient mapping between user defined data and codeword sequences. PDF417 has three data compaction modes:

— Text Compaction mode (see 5.4.1.5);

— Byte Compaction mode (see 5.4.3);

— Numeric Compaction mode (see 5.4.4).

A given string of data bytes may be represented by different codeword sequences, depending on how the encoder switches between compaction modes and sub-modes. There is no single specified way to encode data in a PDF417 symbol.

900 codewords (0 to 899) are available in each mode for data encodation and other functions within the mode. The remaining 29 codewords are assigned to specific functions (see 5.4.1) independent of the current compaction mode.

PDF417 also supports the Extended Channel Interpretation system, which allows different interpretations of data to be accurately encoded in the symbol (see 5.5).

5.4.1 Function codewords

Codewords 900 to 928 are assigned as function codewords as follows:

— for switching between modes (see 5.4.1.1);

— for enhanced applications using Extended Channel Interpretations (ECIs) (see 5.4.1.2);

— for other enhanced applications (see 5.4.1.3 and 5.4.1.4).

At present codewords 903 to 912, 914 to 917, and 919 are reserved. Table 2 defines the complete list of assigned and reserved function codewords. Their functions are defined in 5.4.1.1 to 5.4.1.5. See 5.4.6 for the treatment of reserved codewords.

Table 2 — Assignments of PDF417 function codewords

Codeword	Function	Refer to subclause
900	mode latch to Text Compaction mode	5.4.1.1
901	mode latch to Byte Compaction mode	5.4.1.1, 5.4.3.1
902	mode latch to Numeric Compaction mode	5.4.1.1
903 to 912	Reserved	
913	mode shift to Byte Compaction mode	5.4.1.1
914 to 917, 919	Reserved	
918	linkage flag to associated linear component, in a composite symbol (other than an EAN.UCC Composite symbol)	5.4.1.5
920	linkage flag to associated linear component, in an EAN.UCC Composite symbol	5.4.1.5
921	reader initialisation	5.4.1.4
922	terminator codeword for Macro PDF control block	5.13
923	sequence tag to identify the beginning of optional fields in the Macro PDF control block	5.13
924	mode latch to Byte Compaction mode (used differently from 901)	5.4.1.1, 5.4.3.1
925 to 927	identifier for an Extended Channel Interpretation (ECI)	5.5
928	Macro marker codeword to indicate the beginning of a Macro PDF Control Block	5.13

5.4.1.1 Function codewords for mode switching

In one PDF417 symbol it is possible to switch back and forth between modes as often as required. Advice about selecting the appropriate modes is given in 5.4.5.

A Mode Latch codeword may be used to switch from the current mode to the indicated destination mode which stays in effect until another mode switch is explicitly brought into use. Codewords 900 to 902 and 924 are assigned for this purpose. Table 3 defines their function.

The Mode Shift codeword 913 shall cause a temporary switch from Text Compaction mode to Byte Compaction mode. This switch shall be in effect for only the next codeword, after which the mode shall revert to the prevailing sub-mode of the Text Compaction mode. Codeword 913 is only available in Text Compaction mode; its use is described in 5.4.2.4.

Table 3 — Mode Definition and Mode Switching Codewords

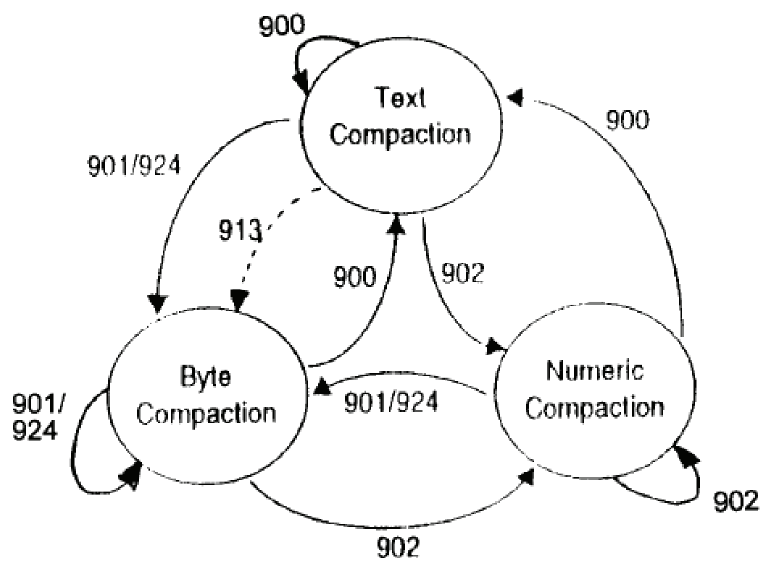
Destination Mode	Mode Latch	Mode Shift
Text Compaction	900	
Byte Compaction	901/924	913
Numeric Compaction	902	

NOTE The table identifies the codeword to be used to switch to the defined mode.

The switching rules between the three modes are defined in Table 4 and shown schematically in Figure 5.

Table 4 — Mode Transition Table, Showing Codewords and Their Function

Original Mode	Destination Mode		
	Text	Byte	Numeric
Text	900 mode latch	913 mode shift 901 mode latch 924 mode latch	902 mode latch
Byte	900 mode latch	901 mode latch 924 mode latch	902 mode latch
Numeric	900 mode latch	901 mode latch 924 mode latch	902 mode latch



Key

— — — mode shift

——— mode latch

Figure 5 — Available Mode Switching

The switching rules into Byte Compaction mode are more fully defined in 5.4.3.1.

5.4.1.2 Function codewords for switching to Extended Channel Interpretations

An ECI codeword can be used to switch to a particular interpretation, which stays in effect until another ECI codeword is explicitly brought into use or until the end of the data. Codewords 925 to 927 are assigned to this function (see 5.5).

5.4.1.3 Function codewords for Macro PDF417

Macro PDF417 symbols (see 5.13) shall use codeword 928 at the start of the Macro PDF417 Control Block. Codewords 922 and 923 are used for special functions in Macro PDF417.

5.4.1.4 Function codeword for reader initialisation

Codeword 921 shall be used to instruct the reader to interpret the data contained within the symbol as programming for reader initialisation. Codeword 921 shall appear as the first codeword after the Symbol Length Descriptor. In the case of a Macro PDF417 initialisation sequence, codeword 921 shall appear in every symbol.

The data contained in an initialisation symbol or sequence of symbols shall not be transmitted by the reader.

5.4.1.5 Function codewords for linkage flags in composite symbols

Codeword 920 shall be used as a linkage flag to signal the presence of an associated EAN.UCC linear component in accordance with ISO/IEC 24723.

Codeword 918 shall be used as a linkage flag to signal the presence of an associated linear component in any other composite symbology.

When used, the 918 or 920 codeword may appear in any position in the symbol. The applicable composite symbology specification may define a specific position of the linkage flag.

Readers supporting the indicated composite application should decode and transmit the data from all components as specified in the relevant composite symbology specification. Readers not supporting the indicated composite application may treat the 918 or 920 codeword as a reserved codeword (see 5.4.6). In addition, readers not supporting the indicated 918 composite application may have an option to ignore the two-dimensional composite component and transmit only the data from the associated linear component.

5.4.2 Text Compaction mode

The Text Compaction mode includes all the printable ASCII characters (i.e. values from 32 to 126) and three ASCII control characters: HT or tab (ASCII value 9), LF or line feed (ASCII value 10), and CR or carriage return (ASCII value 13). The Text Compaction mode also includes various latch and shift characters which are used exclusively within the mode.

The Text Compaction mode encodes up to 2 characters per codeword. The compaction rules for converting data into PDF417 codewords are defined in 5.4.2.2. The sub-mode switches are defined in 5.4.2.3.

5.4.2.1 Text Compaction sub-modes

The Text Compaction mode has four sub-modes:

— Alpha (uppercase alphabetic);

— Lower (lowercase alphabetic);

— Mixed (numeric and some punctuation);

— Punctuation.

Each sub-mode contains 30 characters, including sub-mode latch and shift characters.

The default compaction mode for PDF417 in effect at the start of each symbol shall always be Text Compaction mode Alpha sub-mode (uppercase alphabetic). A latch codeword from another mode to the Text Compaction mode shall always switch to the Text Compaction Alpha sub-mode.

All the characters and their values are defined in Table 5.

Table 5 — Text Compaction Sub-Mode Definition

	Text Compaction Sub-Modes							
Base 30	**Alpha**		**Lower**		**Mixed**		**Punctuation**	
Value	**Char**	**ASCII**	**Char**	**ASCII**	**Char**	**ASCII**	**Char**	**ASCII**
0	A	65	a	97	0	48	;	59
1	B	66	b	98	1	49	<	60
2	C	67	c	99	2	50	>	62
3	D	68	d	100	3	51	@	64
4	E	69	e	101	4	52	[	91
5	F	70	f	102	5	53	\	92
6	G	71	g	103	6	54	]	93
7	H	72	h	104	7	55	_	95
8	I	73	i	105	8	56	‘	96
9	J	74	j	106	9	57	~	126
10	K	75	k	107	&	38	!	33
11	L	76	l	108	CR	13	CR	13
12	M	77	m	109	HT	9	HT	9
13	N	78	n	110	,	44	,	44
14	O	79	o	111	:	58	:	58
15	P	80	p	112	#	35	LF	10
16	Q	81	q	113	-	45	-	45
17	R	82	r	114	.	46	.	46
18	S	83	s	115	$	36	$	36
19	T	84	t	116	/	47	/	47
20	U	85	u	117	+	43	“	34
21	V	86	v	118	%	37	|	124
22	W	87	w	119	*	42	*	42
23	X	88	x	120	=	61	(	40
24	Y	89	y	121	^	94	)	41
25	Z	90	z	122	pl		?	63
26	space	32	space	32	space	32	{	123

al = latch to alpha
as = shift to alpha
ll = latch to lower
ml = latch to mixed
pl = latch to punctuation
ps = shift to punctuation

Table 5 *(continued)*

	Text Compaction Sub-Modes							
Base 30	**Alpha**		**Lower**		**Mixed**		**Punctuation**	
Value	**Char**	**ASCII**	**Char**	**ASCII**	**Char**	**ASCII**	**Char**	**ASCII**
27	ll		as		ll		}	125
28	ml		ml		al		'	39
29	ps		ps		ps		al	

al = latch to alpha
as = shift to alpha
ll = latch to lower
ml = latch to mixed
pl = latch to punctuation
ps = shift to punctuation

NOTE The Char columns above show the default interpretation of ECI 000003 of the byte values shown in the adjacent ASCII columns. Each table entry represents half a codeword, i.e. the value range from 0 to 29 (see 5.4.2.2)

5.4.2.2 Compaction rules for encoding in Text Compaction mode

In Text Compaction mode, pairs of data characters are represented in a single codeword. The values assigned to the data characters are in the range 0 to 29 (i.e. base 30) and are defined in Table 5. For each pair of base 30 values, the first or left value shall be designated the more significant value h, the other shall be designated the less significant value l.

The encoded PDF417 codeword is defined using the following formula:

$$d = h \times 30 + l$$

where d is as defined in Clause 4.

The formula shall also apply to the base 30 values for shifts and latches within the Text Compaction mode. Appropriate latch and shift values shall be used between sub-modes. If the encoding of the character sequence does not result in an even number of base 30 values, see 5.4.2.4 for the specific mechanism to use.

The following example illustrates how compaction is achieved in Text Compaction mode.

EXAMPLE Data to be encoded: PDF417

Table 6 — Example of Text Compaction Encoding

Character Pairs	h	l	$h \times 30 + l$	Codeword
P D	15	3	15 × 30 + 3	453
F ml	5	28	5 × 30 + 28	178
4 1	4	1	4 × 30 + 1	121
7 ps	7	29	7 × 30 + 29	239

NOTE 1 ml (latch to mixed sub-mode) is used to switch to encode the numeric characters.

NOTE 2 ps is used as a pad value in this example, other shift and latch values can be used (see 5.4.2.4)

The data PDF417 is represented by codewords 453, 178, 121, 239

5.4.2.3 Text Compaction sub-mode switching: latch and shift function

Switching from one sub-mode to another within Text Compaction mode shall be through the latch and shift values defined for the sub-mode in effect prior to the switch.

A sub-mode shift shall be used to switch from one Text Compaction sub-mode to another for only one data character. Subsequent codewords revert to the sub-mode being used immediately prior to the shift (except when ps is used as a pad, see 5.4.2.4). The shift functions are as follows:

— ps = shift to punctuation sub-mode;

— as = shift to uppercase alphabetic sub-mode.

A sub-mode latch shall be used to switch from one Text Compaction sub-mode to another, which stays in effect until another latch or shift is explicitly brought into use. The latch functions are as follows:

— al = latch to uppercase alphabetic sub-mode;

— ll = latch to lowercase alphabetic sub-mode;

— ml = latch to mixed (numeric and other punctuation) sub-mode;

— pl = latch to punctuation sub-mode.

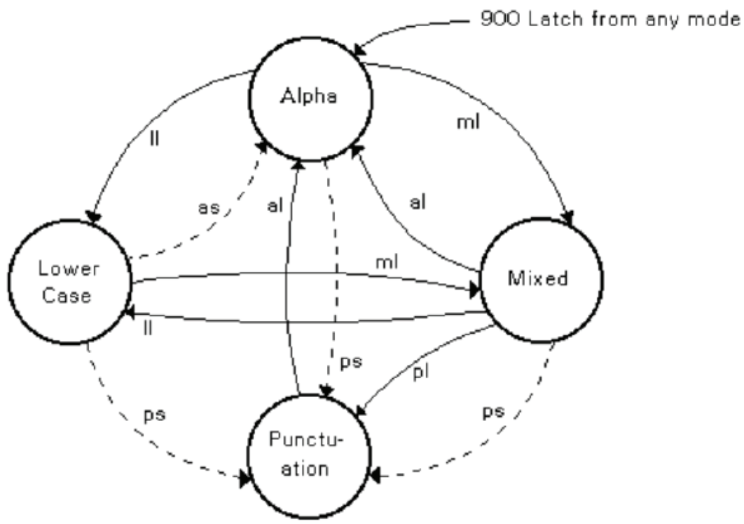
A limited set of latch and shift functions is available within each Text Compaction sub-mode. Those which are available are listed in Table 5. Table 7 shows the transition table between Text Compaction sub-modes; Figure 6 shows this schematically.

NOTE A sub-mode latch may be followed by another sub-mode latch or sub-mode shift; but a sub-mode shift may not be followed by either a sub-mode shift or sub-mode latch.

Table 7 — Text Compaction sub-mode transition table

Original Sub-Mode	Destination Sub-Mode			
	Alpha	Lower	Mixed	Punctuation
Alpha		ll	ml	ps
Lower	as		ml	ps
Mixed	al	ll		ps pl
Punctuation	al			

JA3103



Key

————————	sub-mode latch
_ _ _ _ _ _ _ _ _	sub-mode shift
ll	latch to lower case sub-mode
ps	shift to punctuation sub-mode
ml	latch to mixed sub-mode
as	shift to alpha sub-mode
al	latch to alpha sub-mode
pl	latch to punctuation sub-mode

Figure 6 — Text Compaction Sub-Mode Switching

5.4.2.4 Mechanisms for using a pad in Text Compaction mode

If the Text Compaction character sequence does not result in an even number of base 30 values, a pad shall be added to the end of the character sequence. An example is illustrated in Table 6. As there are no specific null functions in Text Compaction mode, the sub-mode shift and latch shall be used in accordance with the mechanisms defined for the following cases.

The cases are as follows:

a) If the character sequence continues to the end of the data, or the Text Compaction mode character sequence is followed by latching to another compaction mode, then the pad can be any of the sub-mode shifts or sub-mode latches.

b) If the Text Compaction mode character sequence is followed by a byte shift (codeword 913) to encode a single Byte Compaction mode character, two mechanisms can be used depending on the Text Compaction sub-mode being used prior to the Byte Compaction shift:

   1) If the Text Compaction sub-mode is other than punctuation, then base 30 value 29 (ps) should be used if encodation is intended to revert to the same Text Compaction sub-mode. The decoder shall ignore a ps immediately preceding codeword 913.

   2) If the Text Compaction sub-mode is punctuation, then base 30 value 29 (al) shall be used. The decoder shall not ignore the (al), and therefore will return to the Alpha sub-mode.

5.4.2.5 Switching from Text Compaction mode

Text Compaction mode may be terminated by the end of the symbol, or by any of the following codewords:

— 900 (Text Compaction mode latch);

— 901 (Byte Compaction mode latch);

— 902 (Numeric Compaction mode latch);

— 924 (Byte Compaction mode latch);

— 928 (Beginning of Macro PDF417 Control Block);

— 923 (Beginning of Macro PDF417 Optional Field);

— 922 (Macro PDF417 Terminator).

The last three codewords only occur within the Macro PDF417 Control Block of a Macro PDF417 symbol (see 5.13.1). Text Compaction mode is also affected by the presence of a reserved codeword (see 5.4.6).

If the decoder is in the Text Compaction mode and encounters codeword 913 (Byte Compaction mode shift), it decodes the codeword following codeword 913 as a single binary byte and then returns to the Text Compaction mode. The sub-mode to which the decoder returns is the most-recently-latched sub-mode that was in effect prior to codeword 913; a ps sub-mode shift immediately prior to codeword 913 is ignored.

If the decoder is in the Text Compaction mode and encounters codeword 900 (Text Compaction mode latch), the decoder reinitialises to the Alpha sub-mode.

5.4.3 Byte Compaction mode

The Byte Compaction mode enables a sequence of 8-bit bytes to be encoded into a sequence of codewords. It is accomplished by a Base 256 to Base 900 conversion, which achieves a compaction ratio of six bytes to five codewords (1,2 : 1).

All the characters and their values (0 to 255) are defined in Annex B. This shall be treated as the default graphical and control character interpretation. When ECIs are invoked (see 5.5), this interpretation is defined as ECI 000003 (see 5.5.2).

NOTE In previous PDF417 specifications, the default character set corresponded to ECI 000002 (a code page of the MS-DOS operating system). The interpretation of byte character values below 128 is unchanged and the operation of PDF417 printing and scanning equipment is unaffected. New applications that use byte character values above 127 should assume the ECI 000003 default interpretation for broadest compatibility with current systems. Existing applications utilizing values above 127 may continue to encode and process data as before. Applications that rely upon the prior default interpretation of values above 127 may encode ECI 000002 explicitly if they wish to signal this interpretation.

5.4.3.1 Switching to Byte Compaction mode

When in either Text or Numeric Compaction mode, to switch to Byte Compaction mode, it is necessary to use one of the following codewords:

— **mode latch 924** shall be used when the total number of Byte Compaction characters to be encoded is an integer multiple of 6;

— **mode latch 901** shall be used when the total number of Byte Compaction characters to be encoded is not a multiple of 6;

— **mode shift 913** can be used instead of codeword 901 when a single Byte Compaction character has to be encoded.

5.4.3.2 Compaction rules for encoding a single Byte Compaction character (using mode shift 913)

To encode a single Byte Compaction character, the codeword shall be the decimal value (0 to 255) of the character as defined in Annex B.

5.4.3.3 Compaction rules for encoding longer Byte Compaction character strings (using mode latch 924 or 901)

The following procedure shall be used to encode Byte Compaction character data.

a) Establish the total number of Byte Compaction characters.

b) If a perfect multiple of 6, mode latch 924 shall be used; else mode latch 901 shall be used.

c) Sub-divide the number of Byte Compaction characters into a sequence of 6 characters, from left to right (the most to least significant characters). If less than 6 characters, go to Step 7.

d) Assign the decimal values of the 6 data bytes to be encoded in Byte Compaction mode as b_5 to b_0 (where b_5 is the first data byte).

e) Carry out a base 256 to base 900 conversion to produce a sequence of 5 codewords. Annex C defines an algorithm and illustrates a worked example.

f) Repeat from Step 3 as necessary.

g) For the remaining Byte Compaction characters when mode latch 901 is used, (i.e. when the last group is less than 6 Byte Compaction characters) the codeword(s) shall be the decimal value(s) (0 to 255) of the character(s) as defined in Annex B, the most to the least significant.

NOTE Byte Compaction mode following mode latch 901 assumes that the total number of bytes to be encoded is not a multiple of six. If the number of bytes to be encoded in Byte Compaction mode happens to be an integer multiple of six, then either a 901 or a 924 Byte Compaction Latch shall be encoded, placed at any point in the symbol that would create a correct encodation according to these encodation rules. For example, a 924 codeword as either the first or second codeword would identify the following stream of Byte Compaction mode codewords as encoding a multiple-of-six number of bytes. Alternatively, a 901 could be placed at any position within the Byte Compaction mode codeword stream that would split that stream into two segments, neither of which encodes a multiple-of-six number of bytes.

If additional encodation is required in Text Compaction or Numeric Compaction modes, the appropriate latch characters shall be used (see 5.4.1.1).

5.4.3.4 Switching from Byte Compaction

Byte Compaction mode may be terminated by the end of the symbol, or by any of the following codewords:

— 900 (Text Compaction mode latch);

— 901 (Byte Compaction mode latch);

— 902 (Numeric Compaction mode latch);

— 924 (Byte Compaction mode latch);

— 928 (Beginning of Macro PDF417 Control Block);

— 923 (Beginning of Macro PDF417 Optional Field);

— 922 (Macro PDF417 Terminator).

The last three codewords only occur within the Macro PDF417 Control Block of a Macro PDF417 symbol (see 5.13.1). Byte Compaction mode is also affected by the presence of a reserved codeword (see 5.4.6).

Re-invoking Byte Compaction mode (by using codeword 901 or 924 while in Byte Compaction mode) serves to terminate the previous Byte Compaction mode grouping of 6 Byte Compaction characters as described in 5.4.3.3, and then to start a new grouping. This procedure may be necessary when an ECI assignment number needs to be encoded (see 5.5.3.2).

During the decode process for Byte Compaction mode, the treatment of the final group of codewords differs depending on whether Byte Compaction mode is invoked with codeword 901 or 924.

If Byte Compaction mode is invoked with codeword 924, the total number of codewords within the compaction mode shall be a multiple of five. If this is not the case, the symbol is invalid. All the 5-codeword groups are decoded into 6-byte groups.

If Byte Compaction mode is invoked with codeword 901, the final group of codewords is interpreted directly as one byte per codeword, without compaction. Therefore, if the last group consists of five codewords, the group is interpreted as 5 bytes, rather than 6.

5.4.4 Numeric Compaction mode

The Numeric Compaction mode is a method for base 10 to base 900 data compaction and should be used to encode long strings of consecutive numeric digits. The Numeric Compaction mode encodes up to 2,93 numeric digits per codeword.

5.4.4.1 Latch to Numeric Compaction mode

Numeric Compaction mode may be invoked when in Text Compaction or Byte Compaction modes using mode latch 902.

5.4.4.2 Compaction rules for encoding long strings of consecutive numeric digits

The following procedure shall be used to compact numeric data.

a) Divide the string of digits into groups of 44 digits, except for the last group, which may contain fewer.

b) For each group, add the digit 1 to the most significant position to prevent the loss of leading zeros.

EXAMPLE

original data	00246812345678
after step 2	1 00246812345678

NOTE The leading digit 1 is removed in the decode algorithm.

c) Perform a base 10 to base 900 conversion. Annex D defines an algorithm for this and illustrates a worked example.

d) Repeat from Step 2 as necessary.

The following rules can be used to determine the precise number of codewords in Numeric Compaction mode:

— groups of 44 numeric digits compact to 15 codewords;

— for groups of shorter sequences of digits, the number of codewords can be calculated as follows:

Codewords = INT (number of digits/3) +1

EXAMPLE For a 28 digit sequence

INT (28/3) + 1 = 9 + 1 = 10 codewords

5.4.4.3 Switching from Numeric Compaction mode

Numeric Compaction mode may be terminated by the end of the symbol, or by any of the following codewords:

— 900 (Text Compaction mode latch);

— 901 (Byte Compaction mode latch);

— 902 (Numeric Compaction mode latch);

— 924 (Byte Compaction mode latch);

— 928 (Beginning of Macro PDF417 Control Block);

— 923 (Beginning of Macro PDF417 Optional Field);

— 922 (Macro PDF417 Terminator).

The last three codewords only occur within the Macro PDF417 Control Block of a Macro PDF417 symbol (see 5.13.1). Numeric Compaction mode is also affected by the presence of a reserved codeword (see 5.4.6).

Re-invoking Numeric Compaction mode (by using codeword 902 while in Numeric Compaction mode) serves to terminate the current Numeric Compaction mode grouping as described in 5.4.4.2, and then to start a new grouping. This procedure may be necessary when an ECI assignment number needs to be encoded (see 5.5.3.4).

During the decode process for Numeric Compaction mode, the result of the base 900 to base 10 conversion shall result in a number whose most significant digit is a ‘1’. If the base 900 to base 10 conversion does not result in a number beginning with ‘1’, the symbol shall be treated as invalid. The leading ‘1’ is removed to produce the original number.

5.4.5 Advice to select the appropriate compaction mode

All basic implementations for printing and scanning PDF417 symbols shall support the three modes: Text Compaction, Byte Compaction and Numeric Compaction. The default character set for Text Compaction shall be as defined in Table 5; and that for Byte Compaction shall be as defined in Annex B. Text Compaction mode is usually more efficient than Byte Compaction mode for encoding standard ASCII text files because of its better compaction of ASCII character values 9, 10, 13 and 32 to 126.

The Numeric Compaction mode should be used for long numeric strings.

Advice about switching between modes to minimise the number of codewords is provided as an algorithm in Annex N.

5.4.6 Treatment of PDF417 reserved codewords

5.4.6.1 Overview

PDF417 symbols intended for use in open systems should not employ any of the codewords that are listed as reserved (see 5.4.1) in the current edition of this International Standard. However, decoding equipment should support the transmission of reserved codewords using escape sequences as defined in 5.17.4. Decoding equipment may also support an option of treating such symbols as invalid, as would be the case when operating in Basic Channel Mode.

Receiving systems should discard data containing any escape sequences using reserved codewords, unless the system is aware of a new definition for a previously reserved codeword.

5.4.6.2 Making future use of reserved codewords

Any new function codewords, to be defined in future revisions of this International Standard, shall have their encoding rules specified to provide backwards compatibility with pre-existing equipment. Specifically

a) when a new signalling codeword (as opposed to a new compaction mode codeword) is encoded, it shall immediately be followed by an appropriate compaction mode latch so that the subsequent data codewords are interpreted and transmitted as a byte stream, rather than as a series of escaped uninterpreted codewords. This approach will achieve the desired results with decoding equipment conforming with both the original and this PDF417 standard, regardless of whether that equipment employs the original or the new transmission protocol, and

b) at the receiving system, the ECI decoder will process the signal ECIs (i.e. Macro Control Blocks and escaped uninterpreted codewords) before the encodable ECIs (such as character sets). Thus, the encoder should take into account the order of operations as follows:

   1) the Macro Control Block ECIs, if present, will be used to assemble the complete byte stream in the proper order;

   2) the escaped data codewords will be translated by the ECI decoder according to the rules of the new compaction mode or signalling ECI, and the resulting data bytes will be inserted into their proper place within the byte stream;

   3) finally, the character set and other encodable ECIs will be applied to the resulting byte stream.

5.5 Extended Channel Interpretation

The Extended Channel Interpretation (ECI) protocol allows the output data stream to have interpretations different from that of the default character set. The ECI protocol is defined consistently across a number of symbologies, including PDF417. ECIs are assigned by AIM Global, Inc.

NOTE Originally, a symbology specific scheme called Global Label Identifiers (GLIs) was defined for PDF417. Encoding and decoding ECIs is identical to earlier specifications for PDF417 GLIs. However, the transmission protocol for decoded messages according to earlier PDF417 specifications for GLIs is different from the transmission protocol for ECIs. There are also differences with respect to the use of interpretive ECIs with Macro PDF417. This International Standard permits the use of the earlier and current protocols in such a way that old and new equipment can continue to co-exist.

Five broad types of interpretations are supported in PDF417:

a) character sets (or code pages);

b) general purpose interpretations such as data encryption and data compression (as distinct from the compaction modes of the symbology);

c) user defined interpretations for closed systems;

d) transmission of control information for Macro PDF417;

e) transmission of uninterpreted PDF417 codewords.

Transmission of the Extended Channel Interpretation protocol is fully specified in the AIM International standard ITS/04-001, Part 1. The protocol provides a consistent method to specify particular interpretations of byte values before printing and after decoding.

The Extended Channel Interpretation (ECI) is identified by a 6-digit number which is encoded in the PDF417 symbol by one of three specific codewords followed by one or two codewords (see 5.5.1). A specific ECI may be invoked anywhere in the encoded message subject to the rules of the compaction modes (see 5.5.3).

The ECI protocol can only be used with decoders enabled to transmit the symbology identifier (see 5.17.5). Decoders that are not enabled to transmit the symbology identifier cannot reliably convey the escape sequences from any symbol containing an ECI.

5.5.1 Encoding the ECI assignment number

An ECI can be invoked anywhere in the data stream, subject to the conditions defined in 5.5.3. Once an ECI has been invoked, switching may take place between any of the compaction modes. The compaction mode used is determined strictly by the 8-bit data values being encoded and does not depend on the ECI in force. For example, a sequence of values in the range 48 to 57 (decimal) would be most efficiently encoded in Numeric Compaction mode even if the sequence were not to be interpreted as numbers.

The ECI assignment number is encoded in one of the three ECI codeword sequences, which begin with the codewords 927, 926 or 925. One or two additional codewords are used to encode the ECI assignment number. The encodation rules are defined in Table 8.

Table 8 — Encoding ECI assignment numbers

ECI assignment number	Codeword sequence	Codewords	Ranges
000000 to 000899	C_0	927	
	C_1	ECI_no	C_1 = (0 to 899)
000900 to 810899	C_0	926	
	C_1	ECI_no div 900 - 1	C_1 = (0 to 899)
	C_2	ECI_no mod 900	C_2 = (0 to 899)
810900 to 811799	C_0	925	
	C_1	ECI_no - 810 900	C_1 = (0 to 899)

There are 811 800 possible ECI assignment numbers available in PDF417.

NOTE The encodation method is identical to the GLI scheme incorporated in the original AIM USA (1994) and AIM Europe (1994) PDF417 specifications.

The following example illustrates the encodation:

EXAMPLE ECI = 013579

Codewords: [926] [(13 579 div 900) - 1] [13 579 mod 900]

= [926][15 - 1] [79]

= [926] [14] [79]

5.5.2 Pre-assigned and default Extended Channel Interpretations

The following ECIs, ECI 000000 to ECI 000003, have been pre-assigned to be backwards compatible with existing symbology specifications, including PDF417.

— ECI 000000 (equates to original GLI 0) represents the default encodation scheme of encoders compliant with the original PDF417 standards.

— ECI 000001 (equates to original GLI 1) represents the GLI encodation scheme of a number of symbologies with characters 0 to 127 being identical to those of ISO/IEC 646:1991, International Reference Version (equivalent to ANSI X3.4) and characters 128 to 255 being identical to those values of ISO/IEC 8859-1.

NOTE ECI 000000 (equivalent to GLI 0) and ECI 000001 (equivalent to GLI 1) require a return-to-GLI 0 logic at the beginning of each encoded symbol of a Macro PDF417 set of symbols. This protocol is not adopted for other Extended Channel Interpretations.

— ECI 000002 has an equivalent code table to ECI 000000, without the return-to-GLI 0 logic.

— ECI 000003 has an equivalent code table to ECI 000001, without the return-to-GLI 0 logic. ECI 000003 is the default encodation scheme for encoders fully compliant with this edition of this standard.

ECI 000000 and ECI 000001 shall not be encoded in the same PDF417 symbol or Macro PDF417 symbol set as other ECIs, except for user defined ECIs. ECI 000002 and ECI 000003 provide the compatible alternatives to ECI 000000 and ECI 000001 respectively. ECI 000000 and ECI 000001 should not be used in new applications.

5.5.3 Encoding ECI sequences within compaction modes

The general encodation principle is that ECIs are applied to the source data byte stream (to signal various interpretations) producing a modified byte stream that is encoded into PDF417 symbols using the symbology's compaction modes for efficiency. The ECI encoding, and symbology specific compaction, form two independent logical layers of the process.

Although ECI assignments and compaction modes may generally be intermixed, some combinations can produce illogical or ambiguous behaviour. The following clauses define how ECIs may be incorporated without ambiguity by specifying the valid placements of ECI escape sequences.

5.5.3.1 ECIs and Text Compaction mode

An ECI escape sequence may be placed anywhere within Text Compaction mode. The sub-mode invoked immediately prior to the ECI escape sequence is preserved for the encodation immediately after it. Thus, sub-mode latches and shifts are preserved across an ECI escape sequence and thus a sub-mode shift immediately before an ECI escape sequence is not ignored.

5.5.3.2 ECIs and Byte Compaction mode using mode latch 924 and 901

If encoding in Byte Compaction mode using mode latch 924, an ECI escape sequence may be positioned by an encoder immediately following codeword 924, or at any 5-codeword boundary thereafter. This is necessary to provide an unambiguous position in the decoded byte stream for the decoder to place the escape sequence.

If the decoder is in the 924 version of Byte Compaction mode and finds an ECI escape sequence following a 5-codeword group, it shall output the six data bytes associated with the codewords before the escape sequence, output the escape sequence, and then continue collecting codewords for decoding in Byte Compaction mode. If the decoder encounters an ECI escape sequence at other than these prescribed locations, it shall treat the symbol as invalid.

If encoding in Byte Compaction mode using mode latch 901, an ECI escape sequence may be positioned

— immediately following codeword 901,

— immediately after any set of five codewords encoding six bytes, and

— immediately after any of the trailing single-byte codewords at the end of the sequence.

NOTE The decoder cannot assume that, just because the ECI escape sequence follows a set of five codewords, the five codewords encode six bytes, since an input stream of length 6N+5 (where N is an integer) will have a final set of five codewords that encode only five bytes, one byte per codeword. The decoder must, therefore, scan forward in the symbol past the ECI escape sequence to determine where the 901 mode terminates, as defined in 5.4.3.4. Based on this information, it can then determine how the group of five codewords have been encoded.

Figure 7 illustrates valid locations for ECI escape sequences when encoding in Byte Compaction mode. If the decoder encounters an ECI escape sequence within the 5-codeword group, it shall treat the symbol as invalid.

[901] ♦ □ □ □ □ □ ♦ □ □ □ □ □ ♦ □ ♦ □ ♦ □ ♦ □ ♦ □ ♦

[924] ♦ □ □ □ □ □ ♦ □ □ □ □ □ ♦

5 codeword group 5 codeword group

Key

□ byte compaction mode codeword

◆ valid location for ECI escape sequence

Figure 7 — Valid Locations for ECI Escape Sequences in Byte Compaction Mode

5.5.3.3 ECIs and Byte Compaction using mode shift 913

If encoding in Byte Compaction mode using mode shift 913, an ECI escape sequence may be placed

— immediately preceding codeword 913,

— immediately following codeword 913, and

— immediately following the codeword after codeword 913.

In the first two cases, the ECI escape sequence is output before the encoded byte, while in the last case, the escape sequence is output following the encoded byte.

5.5.3.4 ECIs and Numeric Compaction mode

An ECI escape sequence shall not be placed within a group of codewords being processed through the base 10 to base 900 conversion as defined in 5.4.4.2. It may only be placed within a Numeric Compaction mode region at a boundary between (the typically) 15-codeword groups. This is necessary to provide an unambiguous position in the decoded digit stream for the decoder to place the escape sequence.

Thus, an ECI escape sequence may only be placed

— immediately after codeword 902,

— after the 15th codeword,

— after the 30th codeword, and

— etc.

If the encoder needs to place an ECI escape sequence at a location that does not result in a multiple of 15 codewords, it shall treat the numeric block before the ECI as a complete entity, as defined in 5.4.4.2 step 2. It shall re-invoke the Numeric Compaction mode by placing another codeword 902 in the stream followed by the ECI escape sequence.

If the decoder finds an ECI escape sequence on one of the boundary points defined above, it shall emit the data bytes associated with the codewords before the escape sequence (if any), then emit the escape sequence, and then continue collecting codewords for decoding in Numeric Compaction mode. If the decoder encounters an ECI escape sequence at other than the prescribed locations, it shall treat the symbol as invalid.

5.5.3.5 Combining ECIs

Two or more ECI escape sequences (e.g. assignment numbers) may be placed at any point where one ECI can be validly located; providing that no codewords, other than those used to encode the ECI escape sequence, are placed between them.

5.5.4 Post-decode protocol

The protocol for transmitting ECI data shall be as defined in 5.17.2. When transmitting ECIs, symbology identifiers (see 5.17.5) shall be fully implemented and the appropriate symbology identifier shall be transmitted as a preamble.

5.6 Determining the codeword sequence

The encoding process generates a sequence of codewords defined as:

$d_{n-1} \ldots d_0$

where

d is the data codeword including the Symbol Length Descriptor and all function codewords;

n is the total number of data (and pad) codewords including the Symbol Length Descriptor but excluding the error correction codewords.

The Symbol Length Descriptor shall be the first data codeword and designated d_{n-1}. Its value shall be equal to the total number of data codewords $\boldsymbol{n}$; this count shall include the Symbol Length Descriptor itself and thus shall be in the range of 1 to 926.

During the encoding process, sequences of codewords will be established. Like the original data itself, the most significant data shall appear first, for example, textual and numeric data reads from the left to the right. The sequence of codewords shall be that the most significant data codeword containing encoded data is the one designated d_{n-2}. The final data (or pad) codeword is the one designated d_0.

The process used to determine the symbol matrix of rows and columns (see 5.9.2) can require the addition of trailing pad codewords to the end of the data codeword sequence.

5.7 Error detection and correction

Each PDF417 symbol contains at least two error correction codewords. The Error Correction codewords provide capability for both error detection and correction.

5.7.1 Error correction level

The error correction level for a PDF417 symbol is selectable at the time of symbol creation. Table 9 shows the number of error correction codewords for each error correction level.

Table 9 — Error Correction Levels and Error Correction Codewords

Error Correction Level	Total Number of Error Correction Codewords
0	2
1	4
2	8
3	16
4	32
5	64
6	128
7	256
8	512

5.7.2 Error correction capacity

Error correction can be used to compensate for defects in the label and misreads during the decode procedure. For any given error correction level, a particular number of error correction codewords is incorporated into the PDF417 symbol. The error correction codeword algorithm used allows two types of error to be recovered:

— an **erasure**, which is a missing or undecodable codeword at a known position;

— a **substitution error**, which is an erroneously decoded codeword at an unknown position.

The error correction scheme requires one error correction codeword to rectify an erasure and two to recover a substitution error. Thus a given error correction level can rectify any combination of substitution errors and erasures which satisfy the following equations:

$$l + 2f \leq 2^{s+1} - 2$$

where l, f and s are as defined in 4.1.

However, if most of the error correction capacity is used to correct erasures, the possibility of undetected errors is increased. For this reason, whenever there are fewer than 4 errors corrected (except when $s = 0$), the error correction capacity should be reduced as follows:

$$l + 2f \leq 2^{s+1} - 3$$

where l, f and s are as defined in 4.1.

EXAMPLE A PDF417 symbol with error correction level 3 has 16 error correction codewords of which up to 14 can be used to correct errors and erasures. They can correct up to 13 erasures or 7 substitution errors, or any combination of l erasures and f substitution errors subject to the practical equations above. Table 10 specifies the possible combinations.

Table 10 — Possible Error Correction Combinations for Error Correction Level 3

Recovered Substitution Errors	Recovered Erasures	Determining Equation
0	13 or less	$l + 2f \leq 2^{s+1} - 3$ (number of errors <4)
1	11 or less	
2	9 or less	
3	7 or less	
4	6 or less	$l + 2f \leq 2^{s+1} - 2$ (number of errors ≥4)
5	4 or less	
6	2 or less	
7	0	

5.7.3 Defining the error correction codewords

A two-stage process must be performed to define the error correction codewords.

a) Selecting the error correction level. This is a user or application defined option and is described in Annex E.

b) Generating the error correction codewords. This is to a prescribed set of rules defined in 5.10. The procedures cannot be used until all the data codewords, including pad codewords (see 5.9.2) have been defined.

NOTE The procedures defined in 5.3 to 5.9, 5.13 and 5.14 are of prime interest to users. The more technical procedures defined in 5.10, 5.11 and 5.15 are likely to be achieved electronically and require no user decisions.

5.8 Dimensions

PDF417 symbols should conform with the following dimensions.

5.8.1 Minimum width of a module (X)

This should be defined by the application specification, having due regard to the availability of equipment for the production and reading of symbols and complying with the general requirements of the application.

The X dimension shall be constant throughout a given symbol.

NOTE Current bar code symbol quality measurement standards (e.g. ISO/IEC 15415) do not require absolute dimensional measurements to be taken into account for assessing symbol quality. Non-compliance with any minimum dimension should not therefore, by itself, be a reason for rejection of a symbol under these standards.

5.8.2 Row height (Y)

For symbols with at least the recommended minimum level of error correction:

$$Y \geq 3X$$

For symbols with less than the recommended minimum level of error correction, the row height may be increased, particularly when the X dimension is small. (See Annex E for details of the recommended error correction level).

5.8.3 Quiet zones

— Minimum width of horizontal quiet zone (to the left and right of the PDF417 symbol): $2X$

— Minimum size of vertical quiet zone (above and below the PDF417 symbol): $2X$

5.9 Defining the symbol format

The PDF417 symbol matrix, and the overall size and shape of the symbol, are determined by

a) the module width and aspect ratio, and

b) the number of rows and columns in the symbol matrix.

To create a PDF417 symbol, these parameters are selected through a combination of user inputs, application constraints, and default settings. The selection process can be iterative until the user is satisfied with the resultant format.

5.9.1 Defining the aspect ratio of the module

The aspect ratio of the printed module shall be defined by two dimensions:

— X is the desired dimension of the narrowest bar and narrowest space;

— Y is the desired dimension of the height of each row.

These parameters are defined by the user or application. The major factors that determine the values of these parameters are the resolutions of the printing and scanning systems used in the application. These points are discussed in 5.14.

5.9.2 Defining the symbol matrix of rows and columns

There are several factors which need to be considered in order to determine the symbol matrix, i.e. the number of rows *r* and the number of columns *c*:

— the amount and type of data to be encoded;

— the basic rules of the symbology which, for example, determine the limits on the number of rows and columns (see 5.2.1 and 5.2.2);

— the physical space available to print the symbol;

— the fact that longer rows result in the use of less symbol overhead (start and stop characters, row indicators and space for quiet zones);

— the fact that the length of the row (including the quiet zones) must be less than the length of the scan line prescribed or implied by the application;

— the type of scanner, which may determine the overall aspect ratio of the symbol;

— the selected level of error correction.

For many applications, the allowable width of the symbol is the primary constraint, and the symbol matrix can be directly determined by fixing the number of columns. Annex O provides more precise guidelines which should be used to define the symbol matrix.

After the source data has been encoded using the selected compaction modes, the number of source data codewords ***m*** (prior to the addition of the Symbol Length Descriptor and any pad codewords) is known. Once the number of rows and columns, and the error correction level, have been selected, the total number of data codewords ***n*** is calculated as:

$$n = c \times r - k$$

where *c*, *k*, *n* and *r* are as defined in 4.1.

The matrix can result in a situation where the number of rows and columns requires the use of pad codewords (by convention using value **900**). This occurs when:

$$n > m + 1$$

where *m* and *n* are as defined in 4.1.

The Symbol Length Descriptor shall be set to the value ***n*** determined above, thus:

$$d_{n-1} = n = c \times r - k$$

The number of pad codewords required is $(n - m) - 1$.

The pad codewords should have the value 900 and shall be placed in the least significant positions of the data codeword sequence, i.e. to the right of the least-significant source data codeword (but before the Macro PDF417 Control Block, if present). An example of this process is given below. Apart from the insertion of the Symbol Length Descriptor and any pad codewords, the codeword sequence shall remain identical to the one originally generated when encoding the source data.

EXAMPLE

let *m* = 246, *c* = 12, *r* = 24, and *k* = 32, then *n* = (*c* x *r*) - *k* = (12 x 24) - 32 = 256.

NOTE The notation is as defined above.

The value of the Symbol Length Descriptor is *n* = 256.

The number of pad codewords = $(n - m) - 1 = 256 - 246 - 1 = 9$. In this example, the data codewords (before padding) begin with a latch to Numeric Compaction mode (Codeword 902), and end with codeword 423, and the pads all use codeword 900. The addition of the Symbol Length Descriptor and pads is shown below:

Original data codeword sequence:		d_{m-1}	...	d_0			
Codewords:		902	...	423			
Padded data codeword sequence:	d_{n-1}	d_{n-2}	...	d_9	d_8	...	d_0
Codewords:	256	902	...	423	900	...	900

5.10 Generating the error correction codewords

The error correction codewords shall be generated using the procedure defined below. They are calculated on the basis of the values of all the data codewords including the Symbol Length Descriptor and any pad codewords. The codeword sequence is defined as:

$$d_{n-1}, d_{n-2}, \ldots d_0$$

where d_{n-1} is the Symbol Length Descriptor.

The symbol data polynomial is:

$$d(x) = d_{n-1}x^{n-1} + d_{n-2}x^{n-2} + \ldots + d_1 x + d_0$$

The following describes mathematically how the error correction codewords shall be computed for a given stream of data and a selected error correction level. All the arithmetic shall be done in modulo 929.

The error correction codewords are the complement of coefficients of the remainder resulting from dividing the symbol data polynomial $d(x)$ multiplied by x^k by the generator polynomial $g(x)$. Negative values are mapped into the Galois Field GF (929) by adding 929 until the value is ≥0.

The following generator polynomial shall be used to calculate coefficients for k error correction codewords required for the error correction level:

$$g_k(x) = (x-3)(x-3^2)(x-3^3)\ldots(x-3^k)$$

$$= \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \ldots + \alpha_{k-1} x^{k-1} + x^k$$

where

$g_k(x)$ is the generator polynomial and x is the unknown variable;

k is the total number of error correction codewords;

α_j is the coefficient of powers of x produced by the generator polynomial $g_k(x)$

An example for calculating the coefficients is given in Annex Q.

Annex F contains all the coefficient values necessary to encode a PDF417 symbol of any error correction level.

The error correction codewords shall be calculated according to the algorithm defined below using the following notation:

d_i is the data codeword $d_{n-1} \dots d_0$;

E_j is the error correction codeword $E_{k-1} \dots E_0$;

α_j is the coefficient of powers of x taken from the generator polynomial (see above for an explanation and Annex F for the values);

t_1, t_2, t_3 are the temporary variables.

The algorithm is as follows.

a) Identify the data codeword sequence $d_{n-1}, d_{n-2} \dots d_0$.

b) Initialise error correction codewords $E_0, \dots, E_{k-1}$ to value = 0.

c) For each data codeword $d_i = d_{n-1} \dots d_0$:

BEGIN

$t_1 = (d_i + E_{k-1}) \bmod 929$

For each error correction codeword $E_j = E_{k-1} \dots E_1$:

BEGIN

$t_2 = (t_1 \times \alpha_j) \bmod 929$

$t_3 = 929 - t_2$

$E_j = (E_{j-1} + t_3) \bmod 929$

END

$t_2 = (t_1 \times \alpha_0) \bmod 929$

$t_3 = 929 - t_2$

$E_0 = t_3 \bmod 929$

END

d) For each error correction codeword, $E_j = E_0 \dots E_{k-1}$, calculate the complement:

BEGIN

If E_j not equal to 0

$E_j = 929 - E_j$

END

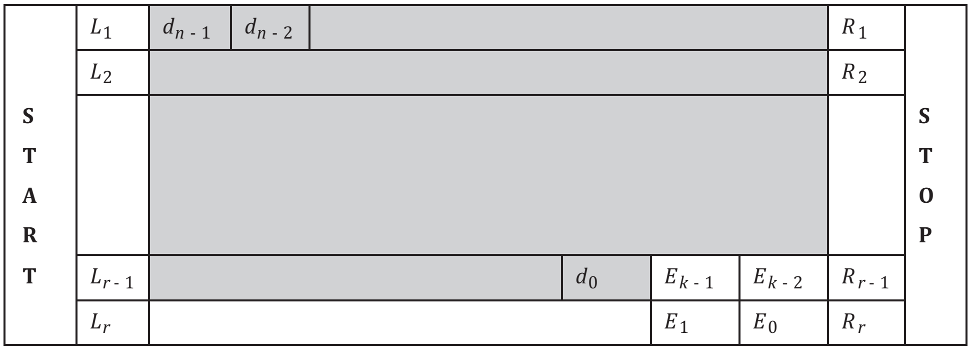
An example of calculating the error correction codewords is given in Annex Q.

An alternative procedure for generating the error correction codewords, using a division circuit, is given in Annex R.

5.11 Low level encodation

Low level encoding converts the codewords into their corresponding symbol characters (bar-space sequence) given that the symbol matrix has been fixed.

Figure 8 illustrates schematically for a PDF417 symbol the corresponding position of each data codeword, error correction codeword and row indicators.



Key

L_r	left row indicator
R_r	right row indicator
shaded area	data codeword area
Unshaded area under the codeword area	for error correction codewords

Figure 8 — Typical PDF417 Symbol Schematic Showing the Positioning of Codewords

5.11.1 Clusters

PDF417 uses a system of local row discrimination to detect row-to-row transitions.

The set of codewords is represented in each of three clusters. Cluster numbers 0, 3 and 6 are used. The associated bar-space sequences of each symbol character representing each codeword and cluster are given in Annex A.

To encode the row indicators and codewords, each row shall contain the symbol characters (bar-space patterns) of only one cluster. Row 1 shall use symbol characters from cluster 0, row 2 shall use symbol characters from cluster 3, row 3 shall use symbol characters from cluster 6, row 4 shall use symbol characters from cluster 0 and so forth. The cluster sequence 0, 3, 6 shall repeat continually. The cluster number K for any row can be calculated:

$$K = \left[\left(rownumber - 1\right)\text{mod}3\right] \times 3$$

where the rows are numbered 1 to r (as defined in 4.1).

Because any two adjacent rows have different clusters, the decoder can utilise scans that cross rows while decoding a PDF417 symbol.

5.11.2 Determining the symbol matrix

The symbol matrix of rows and columns shall be finally determined by the procedures set out in 5.9.2. This provides the values of r and c.

5.11.3 Determining the values of the left and right row indicators

The row indicators in a PDF417 symbol are codewords which encode several key parameters: the row number (F), the number of rows (r), the number of columns (c) and the error correction level (s). The information shall be spread over three rows and the cycle shall repeat continually. The row number (F) shall be encoded in each row.

5.11.3.1 Left row indicators

Left row indicators shall be calculated as follows:

If $K_F = 0$; $L_F = 30 \times [(F - 1) \text{ div } 3] + (r - 1) \text{ div } 3$

If $K_F = 3$; $L_F = 30 \times [(F - 1) \text{ div } 3] + (s \times 3) + (r - 1) \text{ mod } 3$

If $K_F = 6$; $L_F = 30 \times [(F - 1) \text{ div } 3] + (c - 1)$

where c, F, r, s and K are as defined in 4.1

5.11.3.2 Right row indicators

Right row indicators shall be calculated as follows:

If $K_F = 0$; $R_F = 30 \times [(F - 1) \text{ div } 3] + (c - 1)$

If $K_F = 3$; $R_F = 30 \times [(F - 1) \text{ div } 3] + (r - 1) \text{ div } 3$

If $K_F = 6$; $R_F = 30 \times [(F - 1) \text{ div } 3) + (s \times 3] + (r - 1) \text{ mod } 3$

where c, F, r, s, and K are as defined in 4.1.

5.11.4 Row encoding

In each row, the following symbol characters shall conform with the cluster number:

— left row indicator;

— symbol characters representing data and/or error correction codewords to a number equal to the number of columns;

— right row indicator.

The start and stop characters are constant for all rows.

The symbol shall be encoded row by row, taking c (the number of columns) codewords into each row. The first row shall include the Symbol Length Descriptor in the first column. The last row shall include some or all of the error correction codewords.

5.12 Compact PDF417

Compact PDF417 symbols are an available option. If used, Compact PDF417 shall conform with Annex G.

5.13 Macro PDF417

Macro PDF417 provides a mechanism for the data in a file to be split into blocks and be represented in more than one PDF417 symbol. This mechanism is similar to the Structured Append feature in other symbologies.

Each Macro PDF417 symbol shall contain additional control information to enable the original data file to be properly reconstructed, irrespective of the sequence in which the individual PDF417 symbols are scanned and decoded.

Up to 99 999 individual PDF417 symbols may be used to encode data in Macro PDF417.

Full details of the procedures of Macro PDF417 are given in Annex H.

5.13.1 Compaction modes and Macro PDF417

The Macro PDF417 Control Block has a predefined encoding method, so codeword 928 causes the termination of any compaction mode sequence in the body of the symbol. The Segment Index field shall be encoded in Numeric Compaction mode. Each defined Macro PDF417 optional field has a specific, implied initial compaction mode and sub-mode, and the beginning of a new optional field serves to terminate the compaction mode from the previous field (see H.2.3) and initiates its default mode. Specifically, even if two consecutive optional fields both use the Text Compaction mode, the Alpha sub-mode is reset when codeword 923 is encountered.

5.13.2 ECIs and Macro PDF417

Subject to the constraints defined in 5.5.2, ECIs may occur in the message encoded in a single or Macro PDF417 set of symbols. Any ECI invoked shall apply until the end of the encoded data, or until another ECI is encountered. Thus, the interpretation of the ECI may straddle two or more symbols.

The ECI interpretation(s) in the body of the data codeword stream do not extend into the Macro PDF417 Control Block but resume automatically at the beginning of the next symbol. The Control Block's data is interpreted using the default ECI (000003), unless ECI escape sequences are explicitly encoded in an optional field in the Control Block; the effect of any such ECI is automatically terminated at the end of the field in which it appears.

NOTE When implemented as GLIs according to earlier specifications (e.g. the original AIM USA (1994) and AIM Europe (1994) PDF417 specifications), encodation implies a return to GLI 0 (equivalent to ECI 000000) at the start of each symbol. If it is intended for a GLI 1 to persist into the next symbol, then GLI 1 shall be explicitly encoded at the start of this next symbol. As encoders compliant with these earlier standards will be in use for some time, advice is given in 5.17.6 on how to achieve compatibility with this specification.

5.14 User guidelines

5.14.1 Human readable interpretation

PDF417 symbols are capable of encoding large amounts of data, which means that a human readable interpretation of the data characters may not be practical. As an alternative, descriptive text, rather than literal text, may accompany the symbol. The message may be printed anywhere in the area surrounding the symbol, but should not interfere with the symbol itself nor the quiet zones. Font and character size are not specified by this International Standard, but may be by application standards.

5.14.2 Autodiscrimination capability

PDF417 can be used in an autodiscrimination environment with a number of other symbologies (see S.1).

5.14.3 User-defined application parameters

Application standards shall define parameters of PDF417 symbols specified in this International Standard as variable, as follows:

5.14.3.1 Symbology and dimensional characteristics

Application standards shall specify the following data, symbology and dimensional parameters:

a) the selection and use of Extended Channel Interpretations, if required, to extend data encodation beyond the default interpretations of the basic modes;

b) the volume of data in the symbol, which may be fixed, variable or variable up to a defined maximum;

c) the selection of the error correction level;

d) range of *X*-dimension;

e) range of *Y*-dimension;

f) symbol parameters: the range of permissible aspect ratios and/or whether symbol width or height has a maximum size.

NOTE Additional factors which should be taken into consideration when specifying PDF417 applications are given in Annexes O and S.

5.14.3.2 Test specification

The parameters for the evaluation of symbols shall be defined by specifying a quality grade in accordance with ISO/IEC 15415 in the application standard.

This grade is expressed in the form:

grade/aperture/peak response wavelength

The following example illustrates the types of value which need to be expressed:

1,5/10/660

where

1,5 is the overall symbol quality grade;

10 is the measuring aperture reference number (in this example 0,25 mm diameter);

660 is the peak response wavelength in nanometres.

NOTE ISO/IEC 15415 gives guidance on selection of grading parameters in application specifications. The values appropriate for the application shall be defined in the application standard.

5.14.4 PDF417 symbol quality

PDF417 symbols shall be assessed for quality using the 2D bar code symbol print quality guidelines defined in ISO/IEC 15415 for multi-row symbols with cross-row scanning capability.

5.15 Reference decode algorithm

The reference decode algorithm for PDF417 is defined in Annex J. This reference decode algorithm is the basis for print quality assessment according to ISO/IEC 15415.

5.16 Error detection and error correction procedure

As part of the decode procedure, it is possible to reconstruct the symbol for erasures and substitution errors within the error correction capacity of the symbol. This can be done by using the procedure set out in Annex K.

5.17 Transmitted data

5.17.1 Transmitted data in the basic (default) interpretation

All data codewords shall be translated into user data and transmitted as 8-bit bytes, whether this data is encoded in Text Compaction, Byte Compaction or Numeric Compaction mode. Start and stop characters, row indicators, the Symbol Length Descriptor, mode switching codewords, pad codewords and error correction codewords are not transmitted.

5.17.2 Transmission protocol for Extended Channel Interpretation (ECI)

In systems where ECIs are supported, a symbology identifier prefix shall be used with every transmission (see 5.17.5 and Annex L). Macro PDF417 Control Blocks (if transmitted) shall be treated as part of a control set of escape sequences which operate in conjunction with the ECI transmission protocol (see 5.17.3 and Annex H).

Three codewords (925, 926 and 927) signal the encodation of an ECI value and are decoded as byte values as follows.

a) If the ECI sequence begins with codeword 927:

  1) codeword 927 is transmitted as the escape character 92, which represents reverse solidus (\), or backslash, in the default encodation;

  2) the next codeword is converted into a 6-digit value, by placing leading zeros before the codeword. The 6-digit value is transmitted as the six corresponding byte values in the range, 48 to 57.

  EXAMPLE

  Symbol encodes: [927] [123]

  Data transmission (byte): 92, 48, 48, 48, 49, 50, 51

  ASCII interpretation: \000123

b) If the ECI sequence begins with codeword 926:

  1) codeword 926 is transmitted as escape character 92;

  2) the next two codewords are converted into a 6-digit value, with leading zeros if required, using the following formula:

  [(1st codeword) +1] × 900 + (2nd codeword)

  The 6-digit value is transmitted as the six corresponding byte values in the range, 48 to 57.

  EXAMPLE

  Symbol encodes: [926] [136] [156]

  Data transmission (bytes): 92, 49, 50, 51, 52, 53, 54

  ASCII interpretation: \123456

c) If the ECI sequence begins with codeword 925:

  1) codeword 925 is transmitted as escape character 92;

  2) the next codeword is converted into a 6-digit value by adding the value 810 900 to it. The 6-digit value is transmitted as the six corresponding byte values in the range, 48 to 57.

  EXAMPLE

  Symbol encodes: [925] [456]

  Data transmission (byte): 92, 56, 49, 49, 51, 53, 54

  ASCII interpretation: \811356

The procedure is repeated for each occurrence of Extended Channel Interpretation (ECI).

Application software recognising the 7-byte escape sequence of 92 followed by six bytes (each in the range 48 to 57) should interpret all subsequent characters until the end of the encoded data, or until another single byte 92 is encountered, as being from the ECI defined by the 6-digit sequence.

If the reverse solidus, or other character represented by byte 92 needs to be used as encoded data, transmission shall be as follows. Whenever byte 92 occurs as data, two bytes of that value shall be transmitted; thus a single occurrence is always an escape character and a double occurrence indicates true data.

EXAMPLE

Encoded data: A\\B\C

Transmission: A\\\\B\\C

5.17.3 Transmitted data for Macro PDF417

The protocol for transmitted data for Macro PDF417 is included in H.6.

5.17.4 Transmission of reserved codewords using the ECI protocol

When operating under the ECI transmission protocol, PDF417 decoders should transmit a reserved codeword escape sequence of six bytes (interpreted as '\CnnnC'), representing escape character (92) followed by 'C' (67), three digits which represent the decimal value of the reserved codeword, followed by another 'C', which terminates the escape sequence in a symbology-independent manner. The data codewords which follow the reserved codeword are not interpreted by the decoder according to any compaction mode, but instead are transmitted as a series of escape sequences representing the codewords using the same 6-byte escape sequence defined earlier in this paragraph. All remaining data codewords are transmitted in this manner, until one of the following is reached:

— the end of the encoded data in the symbol;

— a latch to a recognised compaction mode;

— a Macro PDF417 Control Block function codeword (928, 923, or 922).

Codeword 913 (Byte shift) is only permitted from Text Compaction mode, and thus shall not be part of the codeword stream while in this process of sending escaped uninterpreted codewords.

NOTE This protocol can properly transmit the message syntax of any reserved codeword whose future definition is to provide either a signalling function or to represent a new compaction mode.

5.17.5 Symbology identifier

Once the structure of the data (in terms of Macro PDF417, ECI, etc) has been identified, the appropriate symbology identifier should be added as a preamble to the transmitted data by the decoder. See Annex L for the symbology identifiers which apply to PDF417.

5.17.6 Transmission using older protocols

The introduction of the Extended Channel Interpretation system, common to a number of symbologies, has had an impact on pre-existing symbologies including PDF417. The basic encoding and decoding rules remain identical in this International Standard to those in the original AIM USA (1994) and AIM Europe (1994) PDF417 specifications. Transmission for both ECIs and Macro PDF417 is different in format, but conveys equivalent information.

All new PDF417 decoding equipment and software should conform with this International Standard. However, equipment conforming with the earlier standard will still be in existence for a number of years. Annex M defines the rules which shall be followed when using decoding equipment and software not capable of being compliant with the current ECI and Macro PDF417 symbols. In this way, old and new decoding equipment can continue to co-exist.

Annex A
(normative)

Encoding/decoding table of PDF417 symbol character bar-space sequences

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
0	31111136	51111125	21111155
1	41111144	61111133	31111163
2	51111152	41111216	11111246
3	31111235	51111224	21111254
4	41111243	61111232	31111262
5	51111251	41111315	11111345
6	21111326	51111323	21111353
7	31111334	61111331	31111361
8	21111425	41111414	11111444
9	11111516	51111422	21111452
10	21111524	41111513	11111543
11	11111615	51111521	61112114
12	21112136	41111612	11112155
13	31112144	41112125	21112163
14	41112152	51112133	61112213
15	21112235	61112141	11112254
16	31112243	31112216	21112262
17	41112251	41112224	61112312
18	11112326	51112232	11112353
19	21112334	31112315	21112361
20	11112425	41112323	61112411
21	11113136	51112331	11112452
22	21113144	31112414	51113114
23	31113152	41112422	61113122
24	11113235	31112513	11113163
25	21113243	41112521	51113213
26	31113251	31112612	61113221
27	11113334	31113125	11113262
28	21113342	41113133	51113312
29	11114144	51113141	11113361
30	21114152	21113216	51113411
31	11114243	31113224	41114114
32	21114251	41113232	51114122
33	11115152	21113315	41114213

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
34	51116111	31113323	51114221
35	31121135	41113331	41114312
36	41121143	21113414	41114411
37	51121151	31113422	31115114
38	21121226	21113513	41115122
39	31121234	31113521	31115213
40	41121242	21113612	41115221
41	21121325	21114125	31115312
42	31121333	31114133	31115411
43	11121416	41114141	21116114
44	21121424	11114216	31116122
45	31121432	21114224	21116213
46	11121515	31114232	31116221
47	21121523	11114315	21116312
48	11121614	21114323	11121146
49	21122135	31114331	21121154
50	31122143	11114414	31121162
51	41122151	21114422	11121245
52	11122226	11114513	21121253
53	21122234	21114521	31121261
54	31122242	11115125	11121344
55	11122325	21115133	21121352
56	21122333	31115141	11121443
57	31122341	11115224	21121451
58	11122424	21115232	11121542
59	21122432	11115323	61122113
60	11123135	21115331	11122154
61	21123143	11115422	21122162
62	31123151	11116133	61122212
63	11123234	21116141	11122253
64	21123242	11116232	21122261
65	11123333	11116331	61122311
66	21123341	41121116	11122352
67	11124143	51121124	11122451
68	21124151	61121132	51123113
69	11124242	41121215	61123121
70	11124341	51121223	11123162
71	21131126	61121231	51123212
72	31131134	41121314	11123261
73	41131142	51121322	51123311
74	21131225	41121413	41124113
75	31131233	51121421	51124121

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
76	41131241	41121512	41124212
77	11131316	41121611	41124311
78	21131324	31122116	31125113
79	31131332	41122124	41125121
80	11131415	51122132	31125212
81	21131423	31122215	31125311
82	11131514	41122223	21126113
83	11131613	51122231	31126121
84	11132126	31122314	21126212
85	21132134	41122322	21126311
86	31132142	31122413	11131145
87	11132225	41122421	21131153
88	21132233	31122512	31131161
89	31132241	31122611	11131244
90	11132324	21123116	21131252
91	21132332	31123124	11131343
92	11132423	41123132	21131351
93	11132522	21123215	11131442
94	11133134	31123223	11131541
95	21133142	41123231	61132112
96	11133233	21123314	11132153
97	21133241	31123322	21132161
98	11133332	21123413	61132211
99	11134142	31123421	11132252
100	21141125	21123512	11132351
101	31141133	21123611	51133112
102	41141141	11124116	11133161
103	11141216	21124124	51133211
104	21141224	31124132	41134112
105	31141232	11124215	41134211
106	11141315	21124223	31135112
107	21141323	31124231	31135211
108	31141331	11124314	21136112
109	11141414	21124322	21136211
110	21141422	11124413	11141144
111	11141513	21124421	21141152
112	21141521	11124512	11141243
113	11142125	11125124	21141251
114	21142133	21125132	11141342
115	31142141	11125223	11141441
116	11142224	21125231	61142111
117	21142232	11125322	11142152

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
118	11142323	11125421	11142251
119	21142331	11126132	51143111
120	11142422	11126231	41144111
121	11142521	41131115	31145111
122	21143141	51131123	11151143
123	11143331	61131131	21151151
124	11151116	41131214	11151242
125	21151124	51131222	11151341
126	31151132	41131313	11152151
127	11151215	51131321	11161142
128	21151223	41131412	11161241
129	31151231	41131511	12111146
130	11151314	31132115	22111154
131	21151322	41132123	32111162
132	11151413	51132131	12111245
133	21151421	31132214	22111253
134	11151512	41132222	32111261
135	11152124	31132313	12111344
136	11152223	41132321	22111352
137	11152322	31132412	12111443
138	11161115	31132511	22111451
139	31161131	21133115	12111542
140	21161222	31133123	62112113
141	21161321	41133131	12112154
142	11161511	21133214	22112162
143	32111135	31133222	62112212
144	42111143	21133313	12112253
145	52111151	31133321	22112261
146	22111226	21133412	62112311
147	32111234	21133511	12112352
148	42111242	11134115	12112451
149	22111325	21134123	52113113
150	32111333	31134131	62113121
151	42111341	11134214	12113162
152	12111416	21134222	52113212
153	22111424	11134313	12113261
154	12111515	21134321	52113311
155	22112135	11134412	42114113
156	32112143	11134511	52114121
157	42112151	11135123	42114212
158	12112226	21135131	42114311
159	22112234	11135222	32115113

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
160	32112242	11135321	42115121
161	12112325	11136131	32115212
162	22112333	41141114	32115311
163	12112424	51141122	22116113
164	12112523	41141213	32116121
165	12113135	51141221	22116212
166	22113143	41141312	22116311
167	32113151	41141411	21211145
168	12113234	31142114	31211153
169	22113242	41142122	41211161
170	12113333	31142213	11211236
171	12113432	41142221	21211244
172	12114143	31142312	31211252
173	22114151	31142411	11211335
174	12114242	21143114	21211343
175	12115151	31143122	31211351
176	31211126	21143213	11211434
177	41211134	31143221	21211442
178	51211142	21143312	11211533
179	31211225	21143411	21211541
180	41211233	11144114	11211632
181	51211241	21144122	12121145
182	21211316	11144213	22121153
183	31211324	21144221	32121161
184	41211332	11144312	11212145
185	21211415	11144411	12121244
186	31211423	11145122	22121252
187	41211431	11145221	11212244
188	21211514	41151113	21212252
189	31211522	51151121	22121351
190	22121126	41151212	11212343
191	32121134	41151311	12121442
192	42121142	31152113	11212442
193	21212126	41152121	12121541
194	22121225	31152212	11212541
195	32121233	31152311	62122112
196	42121241	21153113	12122153
197	21212225	31153121	22122161
198	31212233	21153212	61213112
199	41212241	21153311	62122211
200	11212316	11154113	11213153
201	12121415	21154121	12122252

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
202	22121423	11154212	61213211
203	32121431	11154311	11213252
204	11212415	41161112	12122351
205	21212423	41161211	11213351
206	11212514	31162112	52123112
207	12122126	31162211	12123161
208	22122134	21163112	51214112
209	32122142	21163211	52123211
210	11213126	42111116	11214161
211	12122225	52111124	51214211
212	22122233	62111132	42124112
213	32122241	42111215	41215112
214	11213225	52111223	42124211
215	21213233	62111231	41215211
216	31213241	42111314	32125112
217	11213324	52111322	31216112
218	12122423	42111413	32125211
219	11213423	52111421	31216211
220	12123134	42111512	22126112
221	22123142	42111611	22126211
222	11214134	32112116	11221136
223	12123233	42112124	21221144
224	22123241	52112132	31221152
225	11214233	32112215	11221235
226	21214241	42112223	21221243
227	11214332	52112231	31221251
228	12124142	32112314	11221334
229	11215142	42112322	21221342
230	12124241	32112413	11221433
231	11215241	42112421	21221441
232	31221125	32112512	11221532
233	41221133	32112611	11221631
234	51221141	22113116	12131144
235	21221216	32113124	22131152
236	31221224	42113132	11222144
237	41221232	22113215	12131243
238	21221315	32113223	22131251
239	31221323	42113231	11222243
240	41221331	22113314	21222251
241	21221414	32113322	11222342
242	31221422	22113413	12131441
243	21221513	32113421	11222441

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
244	21221612	22113512	62132111
245	22131125	22113611	12132152
246	32131133	12114116	61223111
247	42131141	22114124	11223152
248	21222125	32114132	12132251
249	22131224	12114215	11223251
250	32131232	22114223	52133111
251	11222216	32114231	51224111
252	12131315	12114314	42134111
253	31222232	22114322	41225111
254	32131331	12114413	32135111
255	11222315	22114421	31226111
256	12131414	12114512	22136111
257	22131422	12115124	11231135
258	11222414	22115132	21231143
259	21222422	12115223	31231151
260	22131521	22115231	11231234
261	12131612	12115322	21231242
262	12132125	12115421	11231333
263	22132133	12116132	21231341
264	32132141	12116231	11231432
265	11223125	51211115	11231531
266	12132224	61211123	12141143
267	22132232	11211164	22141151
268	11223224	51211214	11232143
269	21223232	61211222	12141242
270	22132331	11211263	11232242
271	11223323	51211313	12141341
272	12132422	61211321	11232341
273	12132521	11211362	12142151
274	12133133	51211412	11233151
275	22133141	51211511	11241134
276	11224133	42121115	21241142
277	12133232	52121123	11241233
278	11224232	62121131	21241241
279	12133331	41212115	11241332
280	11224331	42121214	11241431
281	11225141	61212131	12151142
282	21231116	41212214	11242142
283	31231124	51212222	12151241
284	41231132	52121321	11242241
285	21231215	41212313	11251133

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
286	31231223	42121412	21251141
287	41231231	41212412	11251232
288	21231314	42121511	11251331
289	31231322	41212511	12161141
290	21231413	32122115	11252141
291	31231421	42122123	11261132
292	21231512	52122131	11261231
293	21231611	31213115	13111145
294	12141116	32122214	23111153
295	22141124	42122222	33111161
296	32141132	31213214	13111244
297	11232116	41213222	23111252
298	12141215	42122321	13111343
299	22141223	31213313	23111351
300	32141231	32122412	13111442
301	11232215	31213412	13111541
302	21232223	32122511	63112112
303	31232231	31213511	13112153
304	11232314	22123115	23112161
305	12141413	32123123	63112211
306	22141421	42123131	13112252
307	11232413	21214115	13112351
308	21232421	22123214	53113112
309	11232512	32123222	13113161
310	12142124	21214214	53113211
311	22142132	31214222	43114112
312	11233124	32123321	43114211
313	12142223	21214313	33115112
314	22142231	22123412	33115211
315	11233223	21214412	23116112
316	21233231	22123511	23116211
317	11233322	21214511	12211136
318	12142421	12124115	22211144
319	11233421	22124123	32211152
320	11234132	32124131	12211235
321	11234231	11215115	22211243
322	21241115	12124214	32211251
323	31241123	22124222	12211334
324	41241131	11215214	22211342
325	21241214	21215222	12211433
326	31241222	22124321	22211441
327	21241313	11215313	12211532

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
328	31241321	12124412	12211631
329	21241412	11215412	13121144
330	21241511	12124511	23121152
331	12151115	12125123	12212144
332	22151123	22125131	13121243
333	32151131	11216123	23121251
334	11242115	12125222	12212243
335	12151214	11216222	22212251
336	22151222	12125321	12212342
337	11242214	11216321	13121441
338	21242222	12126131	12212441
339	22151321	51221114	63122111
340	11242313	61221122	13122152
341	12151412	11221163	62213111
342	11242412	51221213	12213152
343	12151511	61221221	13122251
344	12152123	11221262	12213251
345	11243123	51221312	53123111
346	11243222	11221361	52214111
347	11243321	51221411	43124111
348	31251122	42131114	42215111
349	31251221	52131122	33125111
350	21251411	41222114	32216111
351	22161122	42131213	23126111
352	12161213	52131221	21311135
353	11252213	41222213	31311143
354	11252312	51222221	41311151
355	11252411	41222312	11311226
356	23111126	42131411	21311234
357	33111134	41222411	31311242
358	43111142	32132114	11311325
359	23111225	42132122	21311333
360	33111233	31223114	31311341
361	13111316	32132213	11311424
362	23111324	42132221	21311432
363	33111332	31223213	11311523
364	13111415	41223221	21311531
365	23111423	31223312	11311622
366	13111514	32132411	12221135
367	13111613	31223411	22221143
368	13112126	22133114	32221151
369	23112134	32133122	11312135

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
370	33112142	21224114	12221234
371	13112225	22133213	22221242
372	23112233	32133221	11312234
373	33112241	21224213	21312242
374	13112324	31224221	22221341
375	23112332	21224312	11312333
376	13112423	22133411	12221432
377	13112522	21224411	11312432
378	13113134	12134114	12221531
379	23113142	22134122	11312531
380	13113233	11225114	13131143
381	23113241	12134213	23131151
382	13113332	22134221	12222143
383	13114142	11225213	13131242
384	13114241	21225221	11313143
385	32211125	11225312	12222242
386	42211133	12134411	13131341
387	52211141	11225411	11313242
388	22211216	12135122	12222341
389	32211224	11226122	11313341
390	42211232	12135221	13132151
391	22211315	11226221	12223151
392	32211323	51231113	11314151
393	42211331	61231121	11321126
394	22211414	11231162	21321134
395	32211422	51231212	31321142
396	22211513	11231261	11321225
397	32211521	51231311	21321233
398	23121125	42141113	31321241
399	33121133	52141121	11321324
400	43121141	41232113	21321332
401	22212125	51232121	11321423
402	23121224	41232212	21321431
403	33121232	42141311	11321522
404	12212216	41232311	11321621
405	13121315	32142113	12231134
406	32212232	42142121	22231142
407	33121331	31233113	11322134
408	12212315	32142212	12231233
409	22212323	31233212	22231241
410	23121422	32142311	11322233
411	12212414	31233311	21322241

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
412	13121513	22143113	11322332
413	12212513	32143121	12231431
414	13122125	21234113	11322431
415	23122133	31234121	13141142
416	33122141	21234212	12232142
417	12213125	22143311	13141241
418	13122224	21234311	11323142
419	32213141	12144113	12232241
420	12213224	22144121	11323241
421	22213232	11235113	11331125
422	23122331	12144212	21331133
423	12213323	11235212	31331141
424	13122422	12144311	11331224
425	12213422	11235311	21331232
426	13123133	12145121	11331323
427	23123141	11236121	21331331
428	12214133	51241112	11331422
429	13123232	11241161	11331521
430	12214232	51241211	12241133
431	13123331	42151112	22241141
432	13124141	41242112	11332133
433	12215141	42151211	12241232
434	31311116	41242211	11332232
435	41311124	32152112	12241331
436	51311132	31243112	11332331
437	31311215	32152211	13151141
438	41311223	31243211	12242141
439	51311231	22153112	11333141
440	31311314	21244112	11341124
441	41311322	22153211	21341132
442	31311413	21244211	11341223
443	41311421	12154112	21341231
444	31311512	11245112	11341322
445	22221116	12154211	11341421
446	32221124	11245211	12251132
447	42221132	51251111	11342132
448	21312116	42161111	12251231
449	22221215	41252111	11342231
450	41312132	32162111	11351123
451	42221231	31253111	21351131
452	21312215	22163111	11351222
453	31312223	21254111	11351321

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
454	41312231	43111115	12261131
455	21312314	53111123	11352131
456	22221413	63111131	11361122
457	32221421	43111214	11361221
458	21312413	53111222	14111144
459	31312421	43111313	24111152
460	22221611	53111321	14111243
461	13131116	43111412	24111251
462	23131124	43111511	14111342
463	33131132	33112115	14111441
464	12222116	43112123	14112152
465	13131215	53112131	14112251
466	23131223	33112214	54113111
467	33131231	43112222	44114111
468	11313116	33112313	34115111
469	12222215	43112321	24116111
470	22222223	33112412	13211135
471	32222231	33112511	23211143
472	11313215	23113115	33211151
473	21313223	33113123	13211234
474	31313231	43113131	23211242
475	23131421	23113214	13211333
476	11313314	33113222	23211341
477	12222413	23113313	13211432
478	22222421	33113321	13211531
479	11313413	23113412	14121143
480	13131611	23113511	24121151
481	13132124	13114115	13212143
482	23132132	23114123	14121242
483	12223124	33114131	13212242
484	13132223	13114214	14121341
485	23132231	23114222	13212341
486	11314124	13114313	14122151
487	12223223	23114321	13213151
488	22223231	13114412	12311126
489	11314223	13114511	22311134
490	21314231	13115123	32311142
491	13132421	23115131	12311225
492	12223421	13115222	22311233
493	13133132	13115321	32311241
494	12224132	13116131	12311324
495	13133231	52211114	22311332

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
496	11315132	62211122	12311423
497	12224231	12211163	22311431
498	31321115	52211213	12311522
499	41321123	62211221	12311621
500	51321131	12211262	13221134
501	31321214	52211312	23221142
502	41321222	12211361	12312134
503	31321313	52211411	13221233
504	41321321	43121114	23221241
505	31321412	53121122	12312233
506	31321511	42212114	13221332
507	22231115	43121213	12312332
508	32231123	53121221	13221431
509	42231131	42212213	12312431
510	21322115	52212221	14131142
511	22231214	42212312	13222142
512	41322131	43121411	14131241
513	21322214	42212411	12313142
514	31322222	33122114	13222241
515	32231321	43122122	12313241
516	21322313	32213114	21411125
517	22231412	33122213	31411133
518	21322412	43122221	41411141
519	22231511	32213213	11411216
520	21322511	42213221	21411224
521	13141115	32213312	31411232
522	23141123	33122411	11411315
523	33141131	32213411	21411323
524	12232115	23123114	31411331
525	13141214	33123122	11411414
526	23141222	22214114	21411422
527	11323115	23123213	11411513
528	12232214	33123221	21411521
529	22232222	22214213	11411612
530	23141321	32214221	12321125
531	11323214	22214312	22321133
532	21323222	23123411	32321141
533	13141412	22214411	11412125
534	11323313	13124114	12321224
535	12232412	23124122	22321232
536	13141511	12215114	11412224
537	12232511	13124213	21412232

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
538	13142123	23124221	22321331
539	23142131	12215213	11412323
540	12233123	22215221	12321422
541	13142222	12215312	11412422
542	11324123	13124411	12321521
543	12233222	12215411	11412521
544	13142321	13125122	13231133
545	11324222	12216122	23231141
546	12233321	13125221	12322133
547	13143131	12216221	13231232
548	11325131	61311113	11413133
549	31331114	11311154	12322232
550	41331122	21311162	13231331
551	31331213	61311212	11413232
552	41331221	11311253	12322331
553	31331312	21311261	11413331
554	31331411	61311311	14141141
555	22241114	11311352	13232141
556	32241122	11311451	12323141
557	21332114	52221113	11414141
558	22241213	62221121	11421116
559	32241221	12221162	21421124
560	21332213	51312113	31421132
561	31332221	61312121	11421215
562	21332312	11312162	21421223
563	22241411	12221261	31421231
564	21332411	51312212	11421314
565	13151114	52221311	21421322
566	23151122	11312261	11421413
567	12242114	51312311	21421421
568	13151213	43131113	11421512
569	23151221	53131121	11421611
570	11333114	42222113	12331124
571	12242213	43131212	22331132
572	22242221	41313113	11422124
573	11333213	51313121	12331223
574	21333221	43131311	22331231
575	13151411	41313212	11422223
576	11333312	42222311	21422231
577	12242411	41313311	11422322
578	11333411	33132113	12331421
579	12243122	43132121	11422421

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
580	11334122	32223113	13241132
581	11334221	33132212	12332132
582	41341121	31314113	13241231
583	31341311	32223212	11423132
584	32251121	33132311	12332231
585	22251212	31314212	11423231
586	22251311	32223311	11431115
587	13161113	31314311	21431123
588	12252113	23133113	31431131
589	11343113	33133121	11431214
590	13161311	22224113	21431222
591	12252311	23133212	11431313
592	24111125	21315113	21431321
593	14111216	22224212	11431412
594	24111224	23133311	11431511
595	14111315	21315212	12341123
596	24111323	22224311	22341131
597	34111331	21315311	11432123
598	14111414	13134113	12341222
599	24111422	23134121	11432222
600	14111513	12225113	12341321
601	24111521	13134212	11432321
602	14112125	11316113	13251131
603	24112133	12225212	12342131
604	34112141	13134311	11433131
605	14112224	11316212	11441114
606	24112232	12225311	21441122
607	14112323	11316311	11441213
608	24112331	13135121	21441221
609	14112422	12226121	11441312
610	14112521	61321112	11441411
611	14113133	11321153	12351122
612	24113141	21321161	11442122
613	14113232	61321211	12351221
614	14113331	11321252	11442221
615	14114141	11321351	11451113
616	23211116	52231112	21451121
617	33211124	12231161	11451212
618	43211132	51322112	11451311
619	23211215	52231211	12361121
620	33211223	11322161	11452121
621	23211314	51322211	15111143

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
622	33211322	43141112	25111151
623	23211413	42232112	15111242
624	33211421	43141211	15111341
625	23211512	41323112	15112151
626	14121116	42232211	14211134
627	24121124	41323211	24211142
628	34121132	33142112	14211233
629	13212116	32233112	24211241
630	14121215	33142211	14211332
631	33212132	31324112	14211431
632	34121231	32233211	15121142
633	13212215	31324211	14212142
634	23212223	23143112	15121241
635	33212231	22234112	14212241
636	13212314	23143211	13311125
637	14121413	21325112	23311133
638	24121421	22234211	33311141
639	13212413	21325211	13311224
640	23212421	13144112	23311232
641	14121611	12235112	13311323
642	14122124	13144211	23311331
643	24122132	11326112	13311422
644	13213124	12235211	13311521
645	14122223	11326211	14221133
646	24122231	61331111	24221141
647	13213223	11331152	13312133
648	23213231	11331251	14221232
649	13213322	52241111	13312232
650	14122421	51332111	14221331
651	14123132	43151111	13312331
652	13214132	42242111	15131141
653	14123231	41333111	14222141
654	13214231	33152111	13313141
655	32311115	32243111	12411116
656	42311123	31334111	22411124
657	52311131	23153111	32411132
658	32311214	22244111	12411215
659	42311222	21335111	22411223
660	32311313	13154111	32411231
661	42311321	12245111	12411314
662	32311412	11336111	22411322
663	32311511	11341151	12411413

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
664	23221115	44111114	22411421
665	33221123	54111122	12411512
666	22312115	44111213	12411611
667	23221214	54111221	13321124
668	33221222	44111312	23321132
669	22312214	44111411	12412124
670	32312222	34112114	13321223
671	33221321	44112122	23321231
672	22312313	34112213	12412223
673	23221412	44112221	22412231
674	22312412	34112312	12412322
675	23221511	34112411	13321421
676	22312511	24113114	12412421
677	14131115	34113122	14231132
678	24131123	24113213	13322132
679	13222115	34113221	14231231
680	14131214	24113312	12413132
681	33222131	24113411	13322231
682	12313115	14114114	12413231
683	13222214	24114122	21511115
684	23222222	14114213	31511123
685	24131321	24114221	41511131
686	12313214	14114312	21511214
687	22313222	14114411	31511222
688	14131412	14115122	21511313
689	12313313	14115221	31511321
690	13222412	53211113	21511412
691	14131511	63211121	21511511
692	13222511	13211162	12421115
693	14132123	53211212	22421123
694	24132131	13211261	32421131
695	13223123	53211311	11512115
696	14132222	44121113	12421214
697	12314123	54121121	22421222
698	13223222	43212113	11512214
699	14132321	44121212	21512222
700	12314222	43212212	22421321
701	13223321	44121311	11512313
702	14133131	43212311	12421412
703	13224131	34122113	11512412
704	12315131	44122121	12421511
705	41411114	33213113	11512511

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
706	51411122	34122212	13331123
707	41411213	33213212	23331131
708	51411221	34122311	12422123
709	41411312	33213311	13331222
710	41411411	24123113	11513123
711	32321114	34123121	12422222
712	42321122	23214113	13331321
713	31412114	24123212	11513222
714	41412122	23214212	12422321
715	42321221	24123311	11513321
716	31412213	23214311	14241131
717	41412221	14124113	13332131
718	31412312	24124121	12423131
719	32321411	13215113	11514131
720	31412411	14124212	21521114
721	23231114	13215212	31521122
722	33231122	14124311	21521213
723	22322114	13215311	31521221
724	23231213	14125121	21521312
725	33231221	13216121	21521411
726	21413114	62311112	12431114
727	22322213	12311153	22431122
728	32322221	22311161	11522114
729	21413213	62311211	12431213
730	31413221	12311252	22431221
731	23231411	12311351	11522213
732	21413312	53221112	21522221
733	22322411	13221161	11522312
734	21413411	52312112	12431411
735	14141114	53221211	11522411
736	24141122	12312161	13341122
737	13232114	52312211	12432122
738	14141213	44131112	13341221
739	24141221	43222112	11523122
740	12323114	44131211	12432221
741	13232213	42313112	11523221
742	23232221	43222211	21531113
743	11414114	42313211	31531121
744	12323213	34132112	21531212
745	22323221	33223112	21531311
746	14141411	34132211	12441113
747	11414213	32314112	22441121

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
748	21414221	33223211	11532113
749	13232411	32314211	12441212
750	11414312	24133112	11532212
751	14142122	23224112	12441311
752	13233122	24133211	11532311
753	14142221	22315112	13351121
754	12324122	23224211	12442121
755	13233221	22315211	11533121
756	11415122	14134112	21541112
757	12324221	13225112	21541211
758	11415221	14134211	12451112
759	41421113	12316112	11542112
760	51421121	13225211	12451211
761	41421212	12316211	11542211
762	41421311	11411144	16111142
763	32331113	21411152	16111241
764	42331121	11411243	15211133
765	31422113	21411251	25211141
766	41422121	11411342	15211232
767	31422212	11411441	15211331
768	32331311	62321111	16121141
769	31422311	12321152	15212141
770	23241113	61412111	14311124
771	33241121	11412152	24311132
772	22332113	12321251	14311223
773	23241212	11412251	24311231
774	21423113	53231111	14311322
775	22332212	52322111	14311421
776	23241311	51413111	15221132
777	21423212	44141111	14312132
778	22332311	43232111	15221231
779	21423311	42323111	14312231
780	14151113	41414111	13411115
781	24151121	34142111	23411123
782	13242113	33233111	33411131
783	23242121	32324111	13411214
784	12333113	31415111	23411222
785	13242212	24143111	13411313
786	14151311	23234111	23411321
787	11424113	22325111	13411412
788	12333212	21416111	13411511
789	13242311	14144111	14321123

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
790	11424212	13235111	24321131
791	12333311	12326111	13412123
792	11424311	11421143	23412131
793	13243121	21421151	13412222
794	11425121	11421242	14321321
795	41431211	11421341	13412321
796	31432112	12331151	15231131
797	31432211	11422151	14322131
798	22342112	11431142	13413131
799	21433112	11431241	22511114
800	21433211	11441141	32511122
801	13252112	45111113	22511213
802	12343112	45111212	32511221
803	11434112	45111311	22511312
804	11434211	35112113	22511411
805	15111116	45112121	13421114
806	15111215	35112212	23421122
807	25111223	35112311	12512114
808	15111314	25113113	22512122
809	15111413	35113121	23421221
810	15111512	25113212	12512213
811	15112124	25113311	13421312
812	15112223	15114113	12512312
813	15112322	25114121	13421411
814	15112421	15114212	12512411
815	15113132	15114311	14331122
816	15113231	15115121	13422122
817	24211115	54211112	14331221
818	24211214	14211161	12513122
819	34211222	54211211	13422221
820	24211313	45121112	12513221
821	34211321	44212112	31611113
822	24211412	45121211	41611121
823	24211511	44212211	31611212
824	15121115	35122112	31611311
825	25121123	34213112	22521113
826	14212115	35122211	32521121
827	24212123	34213211	21612113
828	25121222	25123112	22521212
829	14212214	24214112	21612212
830	24212222	25123211	22521311
831	14212313	24214211	21612311

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
832	24212321	15124112	13431113
833	14212412	14215112	23431121
834	15121511	15124211	12522113
835	14212511	14215211	13431212
836	15122123	63311111	11613113
837	25122131	13311152	12522212
838	14213123	13311251	13431311
839	24213131	54221111	11613212
840	14213222	53312111	12522311
841	15122321	45131111	11613311
842	14213321	44222111	14341121
843	15123131	43313111	13432121
844	14214131	35132111	12523121
845	33311114	34223111	11614121
846	33311213	33314111	31621112
847	33311312	25133111	31621211
848	33311411	24224111	22531112
849	24221114	23315111	21622112
850	23312114	15134111	22531211
851	33312122	14225111	21622211
852	34221221	13316111	13441112
853	23312213	12411143	12532112
854	33312221	22411151	13441211
855	23312312	12411242	11623112
856	24221411	12411341	12532211
857	23312411	13321151	11623211
858	15131114	12412151	31631111
859	14222114	11511134	22541111
860	15131213	21511142	21632111
861	25131221	11511233	13451111
862	13313114	21511241	12542111
863	14222213	11511332	11633111
864	15131312	11511431	16211132
865	13313213	12421142	16211231
866	14222312	11512142	15311123
867	15131411	12421241	25311131
868	13313312	11512241	15311222
869	14222411	11521133	15311321
870	15132122	21521141	16221131
871	14223122	11521232	15312131
872	15132221	11521331	14411114
873	13314122	12431141	24411122

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
874	14223221	11522141	14411213
875	13314221	11531132	24411221
876	42411113	11531231	14411312
877	42411212	11541131	14411411
878	42411311	36112112	15321122
879	33321113	36112211	14412122
880	32412113	26113112	15321221
881	42412121	26113211	14412221
882	32412212	16114112	23511113
883	33321311	16114211	33511121
884	32412311	45212111	23511212
885	24231113	36122111	23511311
886	34231121	35213111	14421113
887	23322113	26123111	24421121
888	33322121	25214111	13512113
889	22413113	16124111	23512121
890	23322212	15215111	13512212
891	24231311	14311151	14421311
892	22413212	13411142	13512311
893	23322311	13411241	15331121
894	22413311	12511133	14422121
895	15141113	22511141	13513121
896	25141121	12511232	32611112
897	14232113	12511331	32611211
898	24232121	13421141	23521112
899	13323113	12512141	22612112
900	14232212	11611124	23521211
901	15141311	21611132	22612211
902	12414113	11611223	14431112
903	13323212	21611231	13522112
904	14232311	11611322	14431211
905	12414212	11611421	12613112
906	13323311	12521132	13522211
907	15142121	11612132	12613211
908	14233121	12521231	32621111
909	13324121	11612231	23531111
910	12415121	11621123	22622111
911	51511112	21621131	14441111
912	51511211	11621222	13532111
913	42421112	11621321	12623111
914	41512112	12531131	16311122
915	42421211	11622131	16311221

Codeword	Bar-space sequence		
	Cluster 0 BSBSBSBS	Cluster 3 BSBSBSBS	Cluster 6 BSBSBSBS
916	41512211	11631122	15411113
917	33331112	11631221	25411121
918	32422112	14411141	15411212
919	33331211	13511132	15411311
920	31513112	13511231	16321121
921	32422211	12611123	15412121
922	31513211	22611131	24511112
923	24241112	12611222	24511211
924	23332112	12611321	15421112
925	24241211	13521131	14512112
926	22423112	12612131	15421211
927	23332211	12621122	14512211
928	21514112	12621221	33611111

Annex B
(normative)

The default character set for Byte Compaction mode

<table>
<tr><th>B</th><th>C</th><th>B</th><th>C</th><th>B</th><th>C</th><th>B</th><th>C</th><th>B</th><th>C</th><th>B</th><th>C</th><th>B</th><th>C</th><th>B</th><th>C</th></tr>
<tr><td>0</td><td>NUL</td><td>32</td><td>space</td><td>64</td><td>@</td><td>96</td><td>`</td><td>128</td><td></td><td>160</td><td>NBSP</td><td>192</td><td>À</td><td>224</td><td>à</td></tr>
<tr><td>1</td><td>SOH</td><td>33</td><td>!</td><td>65</td><td>A</td><td>97</td><td>a</td><td>129</td><td></td><td>161</td><td>¡</td><td>193</td><td>Á</td><td>225</td><td>á</td></tr>
<tr><td>2</td><td>STX</td><td>34</td><td>“</td><td>66</td><td>B</td><td>98</td><td>b</td><td>130</td><td></td><td>162</td><td>¢</td><td>194</td><td>Â</td><td>226</td><td>â</td></tr>
<tr><td>3</td><td>ETX</td><td>35</td><td>#</td><td>67</td><td>C</td><td>99</td><td>c</td><td>131</td><td></td><td>163</td><td>£</td><td>195</td><td>Ã</td><td>227</td><td>ã</td></tr>
<tr><td>4</td><td>EOT</td><td>36</td><td>$</td><td>68</td><td>D</td><td>100</td><td>d</td><td>132</td><td></td><td>164</td><td>¤</td><td>196</td><td>Ä</td><td>228</td><td>ä</td></tr>
<tr><td>5</td><td>ENQ</td><td>37</td><td>%</td><td>69</td><td>E</td><td>101</td><td>e</td><td>133</td><td></td><td>165</td><td>¥</td><td>197</td><td>Å</td><td>229</td><td>å</td></tr>
<tr><td>6</td><td>ACK</td><td>38</td><td>&</td><td>70</td><td>F</td><td>102</td><td>f</td><td>134</td><td></td><td>166</td><td>¦</td><td>198</td><td>Æ</td><td>230</td><td>æ</td></tr>
<tr><td>7</td><td>BEL</td><td>39</td><td>‘</td><td>71</td><td>G</td><td>103</td><td>g</td><td>135</td><td></td><td>167</td><td>§</td><td>199</td><td>Ç</td><td>231</td><td>ç</td></tr>
<tr><td>8</td><td>BS</td><td>40</td><td>(</td><td>72</td><td>H</td><td>104</td><td>h</td><td>136</td><td></td><td>168</td><td>¨</td><td>200</td><td>È</td><td>232</td><td>è</td></tr>
<tr><td>9</td><td>HT</td><td>41</td><td>)</td><td>73</td><td>I</td><td>105</td><td>I</td><td>137</td><td></td><td>169</td><td>©</td><td>201</td><td>É</td><td>233</td><td>é</td></tr>
<tr><td>10</td><td>LF</td><td>42</td><td>*</td><td>74</td><td>J</td><td>106</td><td>j</td><td>138</td><td></td><td>170</td><td>ª</td><td>202</td><td>Ê</td><td>234</td><td>ê</td></tr>
<tr><td>11</td><td>VT</td><td>43</td><td>+</td><td>75</td><td>K</td><td>107</td><td>k</td><td>139</td><td></td><td>171</td><td>«</td><td>203</td><td>Ë</td><td>235</td><td>ë</td></tr>
<tr><td>12</td><td>FF</td><td>44</td><td>,</td><td>76</td><td>L</td><td>108</td><td>l</td><td>140</td><td></td><td>172</td><td>¬</td><td>204</td><td>Ì</td><td>236</td><td>ì</td></tr>
<tr><td>13</td><td>CR</td><td>45</td><td>-</td><td>77</td><td>M</td><td>109</td><td>m</td><td>141</td><td></td><td>173</td><td>SHY</td><td>205</td><td>Í</td><td>237</td><td>í</td></tr>
<tr><td>14</td><td>SO</td><td>46</td><td>.</td><td>78</td><td>N</td><td>110</td><td>n</td><td>142</td><td></td><td>174</td><td>®</td><td>206</td><td>Î</td><td>238</td><td>î</td></tr>
<tr><td>15</td><td>SI</td><td>47</td><td>/</td><td>79</td><td>O</td><td>111</td><td>o</td><td>143</td><td></td><td>175</td><td>¯</td><td>207</td><td>Ï</td><td>239</td><td>ï</td></tr>
<tr><td>16</td><td>DLE</td><td>48</td><td>0</td><td>80</td><td>P</td><td>112</td><td>p</td><td>144</td><td></td><td>176</td><td>°</td><td>208</td><td>Ð</td><td>240</td><td>ð</td></tr>
<tr><td>17</td><td>DC1</td><td>49</td><td>1</td><td>81</td><td>Q</td><td>113</td><td>q</td><td>145</td><td></td><td>177</td><td>±</td><td>209</td><td>Ñ</td><td>241</td><td>ñ</td></tr>
<tr><td>18</td><td>DC2</td><td>50</td><td>2</td><td>82</td><td>R</td><td>114</td><td>r</td><td>146</td><td></td><td>178</td><td>²</td><td>210</td><td>Ò</td><td>242</td><td>ò</td></tr>
<tr><td>19</td><td>DC3</td><td>51</td><td>3</td><td>83</td><td>S</td><td>115</td><td>s</td><td>147</td><td></td><td>179</td><td>³</td><td>211</td><td>Ó</td><td>243</td><td>ó</td></tr>
<tr><td>20</td><td>DC4</td><td>52</td><td>4</td><td>84</td><td>T</td><td>116</td><td>t</td><td>148</td><td></td><td>180</td><td>´</td><td>212</td><td>Ô</td><td>244</td><td>ô</td></tr>
<tr><td>21</td><td>NAK</td><td>53</td><td>5</td><td>85</td><td>U</td><td>117</td><td>u</td><td>149</td><td></td><td>181</td><td>µ</td><td>213</td><td>Õ</td><td>245</td><td>õ</td></tr>
<tr><td>22</td><td>SYN</td><td>54</td><td>6</td><td>86</td><td>V</td><td>118</td><td>v</td><td>150</td><td></td><td>182</td><td>¶</td><td>214</td><td>Ö</td><td>246</td><td>ö</td></tr>
<tr><td>23</td><td>ETB</td><td>55</td><td>7</td><td>87</td><td>W</td><td>119</td><td>w</td><td>151</td><td></td><td>183</td><td>·</td><td>215</td><td>×</td><td>247</td><td>÷</td></tr>
<tr><td>24</td><td>CAN</td><td>56</td><td>8</td><td>88</td><td>X</td><td>120</td><td>x</td><td>152</td><td></td><td>184</td><td>¸</td><td>216</td><td>Ø</td><td>248</td><td>ø</td></tr>
<tr><td>25</td><td>EM</td><td>57</td><td>9</td><td>89</td><td>Y</td><td>121</td><td>y</td><td>153</td><td></td><td>185</td><td>¹</td><td>217</td><td>Ù</td><td>249</td><td>ù</td></tr>
<tr><td>26</td><td>SUB</td><td>58</td><td>:</td><td>90</td><td>Z</td><td>122</td><td>z</td><td>154</td><td></td><td>186</td><td>º</td><td>218</td><td>Ú</td><td>250</td><td>ú</td></tr>
<tr><td>27</td><td>ESC</td><td>59</td><td>;</td><td>91</td><td>[</td><td>123</td><td>{</td><td>155</td><td></td><td>187</td><td>»</td><td>219</td><td>Û</td><td>251</td><td>û</td></tr>
<tr><td>28</td><td>IS4/FS</td><td>60</td><td><</td><td>92</td><td>\</td><td>124</td><td>|</td><td>156</td><td></td><td>188</td><td>¼</td><td>220</td><td>Ü</td><td>252</td><td>ü</td></tr>
<tr><td>29</td><td>IS3/GS</td><td>61</td><td>=</td><td>93</td><td>]</td><td>125</td><td>}</td><td>157</td><td></td><td>189</td><td>½</td><td>221</td><td>Ý</td><td>253</td><td>ý</td></tr>
<tr><td>30</td><td>IS2/RS</td><td>62</td><td>></td><td>94</td><td>^</td><td>126</td><td>~</td><td>158</td><td></td><td>190</td><td>¾</td><td>222</td><td>Þ</td><td>254</td><td>þ</td></tr>
<tr><td>31</td><td>IS1/US</td><td>63</td><td>?</td><td>95</td><td>_</td><td>127</td><td>DEL</td><td>159</td><td></td><td>191</td><td>¿</td><td>223</td><td>ß</td><td>255</td><td>ÿ</td></tr>
</table>

NOTE This table corresponds to the character set defined in ISO/IEC 8859-1, with the addition of the control characters (byte values 00 – 31) defined in ISO/IEC 646, International Reference Version.

Annex C
(normative)

Byte Compaction mode encoding algorithm

This conversion is used in Byte Compaction mode. It converts six data bytes to five PDF417 data codewords. The conversion equation is:

$$b_5 \times 256^5 + b_4 \times 256^4 + b_3 \times 256^3 + b_2 \times 256^2 + b_1 \times 256^1 + b_0 \times 256^0$$

$$= d_4 \times 900^4 + d_3 \times 900^3 + d_2 \times 900^2 + d_1 \times 900^1 + d_0 \times 900^0$$

where

b is the data byte value as a decimal (0 to 255);

d is the data codeword.

The following algorithm may be used for a base 256 to base 900 conversion.

a) Designate t = temporary variable

b) Calculate $t = b_5 \times 256^5 + b_4 \times 256^4 + b_3 \times 256^3 + b_2 \times 256^2 + b_1 \times 256^1 + b_0 \times 256^0$

c) Calculate each codeword as follows:

For each data codeword $d_i = d_o \ldots d_4$

BEGIN

d_i = t mod 900

t = t div 900

END

EXAMPLE

Encode the Byte Compaction characters $b_5 \ldots b_0$ {231, 101, 11, 97, 205, 2}

Calculate the sum t using the decimal values of the six Byte Compaction characters:

t = 231 × 256^5 + 101 × 256^4 + 11 × 256^3 + 97 × 256^2 + 205 × 256^1

+ 2 × 256^0

= 254 421 168 672 002

Calculate codeword 0

d_0 = 254 421 168 672 002 mod 900 = 302

t = 254 421 168 672 002 div 900 = 282 690 187 413

Calculate codeword 1

d_1 = 282 690 187 413 mod 900 = 213

t = 282 690 187 413 div 900 = 314 100 208

Calculate codeword 2

d_2 = 314 100 208 mod 900 = 208

t = 314 100 208 div 900 = 349 000

Calculate codeword 3

d_3 = 349 000 mod 900 = 700

t = 349 000 div 900 = 387

Calculate codeword 4

d_4 = 387 mod 900 = 387

t = 387 div 900 = 0

The codeword sequence d_4 ... d_0 is 387, 700, 208, 213, 302

Annex D
(normative)

Numeric Compaction mode encoding algorithm

This conversion is used in Numeric Compaction mode. It converts groups of up to 44 consecutive numeric digits to 15 or fewer PDF417 data codewords.

The following algorithm may be used for a base 10 to base 900 conversion.

a) Designate t = temporary value.

b) Set the initial value of t to be the group of up to 44 consecutive numeric digits, preceded by the digit 1.

c) Calculate each codeword as follows:

For each data codeword $d_i = d_o \ldots. d_{n-1}$

BEGIN

d_i	=	t mod 900
t	=	t div 900
If t	=	0, then stop encoding

END

EXAMPLE

Encode the fifteen digit numeric string 000213298174000

Prefix the numeric string with a 1 and set the initial value of

t = 1 000 213 298 174 000

Calculate codeword 0

d_0 = 1 000 213 298 174 000 mod 900 = 200

t = 1 000 213 298 174 000 div 900 = 1 111 348 109 082

Calculate codeword 1

d_1 = 1 111 348 109 082 mod 900 = 282

t = 1 111 348 109 082 div 900 = 1 234 831 232

Calculate codeword 2

d_2 = 1 234 831 232 mod 900 = 632

t = 1 234 831 232 div 900 1 372 034

Calculate codeword 3

d_3 = 1 372 034 mod 900 = 434

t = 1 372 034 div 900 = 1 524

Calculate codeword 4

d_4 = 1 524 mod 900 = 624

t = 1 524 div 900 = 1

Calculate codeword 5

d_5 = 1 mod 900 = 1

t = 1 div 900 = 0

The codeword sequence d_5 ... d_0 is 1, 624, 434, 632, 282, 200

Annex E
(normative)

User selection of error correction level

E.1 Recommended minimum error correction level

The minimum level of error correction level should be as defined in Table E.1.

Table E.1 — Recommended Error Correction Level

Number of Data Codewords	Minimum Error Correction Level
1 to 40	2
41 to 160	3
161 to 320	4
321 to 863	5

As a guide for estimating the number of data codewords from data content in order to use Table E.1, use 1,8 text characters per data codeword in Text Compaction mode, 2,9 digits per data codeword in Numeric Compaction mode and 1,2 bytes per data codeword in Byte Compaction mode.

Higher levels of error correction should be used where significant symbol damage or degradation is anticipated. Lower than recommended error correction levels may be used in closed system applications.

E.2 Other user consideration of the error correction level

The objective in an application standard should be to make use of the features of error correction without sacrificing the data content capacity.

The following factors should be taken into account by the user in selecting an error correction level.

a) The recommended error correction level (see Table E.1) should be followed.

b) Since the maximum number of data codewords per symbol is fixed at 925, large numbers of data codewords limit the maximum level of error correction that can be implemented. More than 415 data codewords precludes Error Correction Level 8. More than 671 data codewords precludes Levels 7 and 8. More than 799 data codewords precludes Levels 6, 7 and 8. More than 863 data codewords precludes Level 5 and therefore is not recommended.

c) Where PDF417 symbols are likely to have missing or totally obliterated codewords, the Error Correction Level may be increased up to Error Correction level 8, or up to a level where the number of error correction codewords fills the maximum sized matrix appropriate for the application.

d) It is preferable to maintain symbol quality rather than to compensate for poor print quality by increasing the error correction level. Instead of adopting a higher error correction level, it may be better to specify a larger X-dimension or particular substrates and materials which can maintain the print quality of the PDF417 symbol.

Annex F
(normative)

Tables of coefficients for calculating PDF417 error correction codewords

Table F.1 — Coefficient table for error correction level 0

j	0	1
α_j	27	917

Table F.2 — Coefficient table for error correction level 1

j	0	1	2	3
α_j	522	568	723	809

Table F.3 — Coefficient table for error correction level 2

j	0	1	2	3	4	5	6	7
α_j	237	308	436	284	646	653	428	379

Table F.4 — Coefficient table for error correction level 3

j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
α_j	274	562	232	755	599	524	801	132	295	116	442	428	295	42	176	65

Table F.5 — Coefficient table for error correction level 4

j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
α_j	361	575	922	525	176	586	640	321	536	742	677	742	687	284	193	517
j	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
α_j	273	494	263	147	593	800	571	320	803	133	231	390	685	330	63	410

Table F.6 — Coefficient table for error correction level 5

j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
α_j	539	422	6	93	862	771	453	106	610	287	107	505	733	877	381	612
j	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
α_j	723	476	462	172	430	609	858	822	543	376	511	400	672	762	283	184
j	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
α_j	440	35	519	31	460	594	225	535	517	352	605	158	651	201	488	502
j	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
α_j	648	733	717	83	404	97	280	771	840	629	4	381	843	623	264	543

Table F.7 — Coefficient table for error correction level 6

j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
α_j	521	310	864	547	858	580	296	379	53	779	897	444	400	925	749	415
j	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
α_j	822	93	217	208	928	244	583	620	246	148	447	631	292	908	490	704
j	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
α_j	516	258	457	907	594	723	674	292	272	96	684	432	686	606	860	569
j	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
α_j	193	219	129	186	236	287	192	775	278	173	40	379	712	463	646	776
j	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
α_j	171	491	297	763	156	732	95	270	447	90	507	48	228	821	808	898
j	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
α_j	784	663	627	378	382	262	380	602	754	336	89	614	87	432	670	616
j	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
α_j	157	374	242	726	600	269	375	898	845	454	354	130	814	587	804	34
j	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
α_j	211	330	539	297	827	865	37	517	834	315	550	86	801	4	108	539

Table F.8 — Coefficient table for error correction level 7

j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
α_j	524	894	75	766	882	857	74	204	82	586	708	250	905	786	138	720
j	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
α_j	858	194	311	913	275	190	375	850	438	733	194	280	201	280	828	757
j	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
α_j	710	814	919	89	68	569	11	204	796	605	540	913	801	700	799	137
j	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
α_j	439	418	592	668	353	859	370	694	325	240	216	257	284	549	209	884
j	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
α_j	315	70	329	793	490	274	877	162	749	812	684	461	334	376	849	521
j	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
α_j	307	291	803	712	19	358	399	908	103	511	51	8	517	225	289	470
j	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
α_j	637	731	66	255	917	269	463	830	730	433	848	585	136	538	906	90
j	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
α_j	2	290	743	199	655	903	329	49	802	580	355	588	188	462	10	134
j	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
α_j	628	320	479	130	739	71	263	318	374	601	192	605	142	673	687	234
j	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
α_j	722	384	177	752	607	640	455	193	689	707	805	641	48	60	732	621
j	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
α_j	895	544	261	852	655	309	697	755	756	60	231	773	434	421	726	528
j	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
α_j	503	118	49	795	32	144	500	238	836	394	280	566	319	9	647	550

Table F.8 *(continued)*

j	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
α_j	73	914	342	126	32	681	331	792	620	60	609	441	180	791	893	754
j	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
α_j	605	383	228	749	760	213	54	297	134	54	834	299	922	191	910	532
j	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
α_j	609	829	189	20	167	29	872	449	83	402	41	656	505	579	481	173
j	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255
α_j	404	251	688	95	497	555	642	543	307	159	924	558	648	55	497	10

Table F.9 — Coefficient table for error correction level 8

j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
α_j	352	77	373	504	35	599	428	207	409	574	118	498	285	380	350	492
j	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
α_j	197	265	920	155	914	299	229	643	294	871	306	88	87	193	352	781
j	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
α_j	846	75	327	520	435	543	203	666	249	346	781	621	640	268	794	534
j	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
α_j	539	781	408	390	644	102	476	499	290	632	545	37	858	916	552	41
j	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
α_j	542	289	122	272	383	800	485	98	752	472	761	107	784	860	658	741
j	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
α_j	290	204	681	407	855	85	99	62	482	180	20	297	451	593	913	142
j	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
α_j	808	684	287	536	561	76	653	899	729	567	744	390	513	192	516	258
j	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
α_j	240	518	794	395	768	848	51	610	384	168	190	826	328	596	786	303
j	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
α_j	570	381	415	641	156	237	151	429	531	207	676	710	89	168	304	402
j	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
α_j	40	708	575	162	864	229	65	861	841	512	164	477	221	92	358	785
j	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
α_j	288	357	850	836	827	736	707	94	8	494	114	521	2	499	851	543
j	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
α_j	152	729	771	95	248	361	578	323	856	797	289	51	684	466	533	820
j	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
α_j	669	45	902	452	167	342	244	173	35	463	651	51	699	591	452	578
j	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
α_j	37	124	298	332	552	43	427	119	662	777	475	850	764	364	578	911
j	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
α_j	283	711	472	420	245	288	594	394	511	327	589	777	699	688	43	408
j	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255
α_j	842	383	721	521	560	644	714	559	62	145	873	663	713	159	672	729

Table F.9 *(continued)*

j	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270	271
α_j	624	59	193	417	158	209	563	564	343	693	109	608	563	365	181	772
j	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287
α_j	677	310	248	353	708	410	579	870	617	841	632	860	289	536	35	777
j	288	289	290	291	292	293	294	295	296	297	298	299	300	301	302	303
α_j	618	586	424	833	77	597	346	269	757	632	695	751	331	247	184	45
j	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319
α_j	787	680	18	66	407	369	54	492	228	613	830	922	437	519	644	905
j	320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335
α_j	789	420	305	441	207	300	892	827	141	537	381	662	513	56	252	341
j	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351
α_j	242	797	838	837	720	224	307	631	61	87	560	310	756	665	397	808
j	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367
α_j	851	309	473	795	378	31	647	915	459	806	590	731	425	216	548	249
j	368	369	370	371	372	373	374	375	376	377	378	379	380	381	382	383
α_j	321	881	699	535	673	782	210	815	905	303	843	922	281	73	469	791
j	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399
α_j	660	162	498	308	155	422	907	817	187	62	16	425	535	336	286	437
j	400	401	402	403	404	405	406	407	408	409	410	411	412	413	414	415
α_j	375	273	610	296	183	923	116	667	751	353	62	366	691	379	687	842
j	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431
α_j	37	357	720	742	330	5	39	923	311	424	242	749	321	54	669	316
j	432	433	434	435	436	437	438	439	440	441	442	443	444	445	446	447
α_j	342	299	534	105	667	488	640	672	576	540	316	486	721	610	46	656
j	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462	463
α_j	447	171	616	464	190	531	297	321	762	752	533	175	134	14	381	433
j	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479
α_j	717	45	111	20	596	284	736	138	646	411	877	669	141	919	45	780
j	480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495
α_j	407	164	332	899	165	726	600	325	498	655	357	752	768	223	849	647
j	496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511
α_j	63	310	863	251	366	304	282	738	675	410	389	244	31	121	303	263

Annex G
(normative)

Compact PDF417

G.1 Description

Compact PDF417 may be used where space considerations are a primary concern and symbol damage is unlikely. In an environment where label damage is unlikely (e.g. an office), the right row indicators may be omitted and the stop pattern may be reduced to one module width bar, as indicated in Figure G.1. This procedure reduces the non-data overhead from 4 codewords per row to 2 codewords per row, with some trade-off in decode performance and robustness, or the ability to withstand noise, damage, degradation, dust etc.

This overhead reduction version is called Compact PDF417, which is fully decoder compatible with standard PDF417.

A Compact PDF417 symbol with fewer than 6 rows encodes the number of columns in only one place, which is not error corrected, and is therefore extremely vulnerable to poor print quality or damage.

NOTE In the original AIM USA (1994) and AIM Europe (1994) PDF417 specifications, the term Truncated PDF417 has been used in a technically synonymous manner. The name Compact PDF417 is preferred to avoid confusion with the more general use of the term 'truncated'.

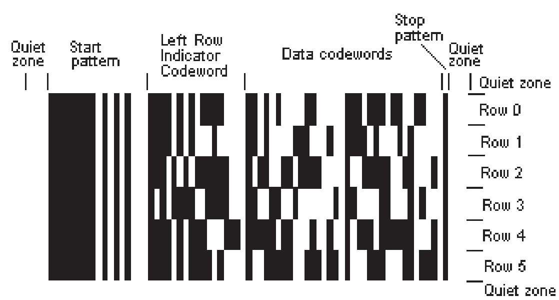


Figure G.1 — Compact PDF417

G.2 Print quality

Although the standard print quality method specified in 5.14.4 is applied to Compact PDF417, the absence of a Stop Pattern (other than the single module bar) requires two exceptions to be made.

The analysis of scan reflectance profiles for the Start and Stop Patterns applies only to the Start Pattern.

For the assessment of Codeword Yield, the requirement that a qualifying scan of the top or bottom row of the symbol (which ISO/IEC 15415 includes the decoding of both Start and Stop Patterns) cannot be applied; instead, as for other rows, the Start Pattern and at least one additional codeword must have been decoded.

Annex H
(normative)

Macro PDF417

H.1 Macro PDF417 overview

Macro PDF417 provides a standard mechanism for creating a distributed representation of files too large to be represented by a single PDF417 symbol. Macro PDF417 symbols differ from ordinary PDF417 symbols in that they contain additional control information in a Macro PDF417 Control Block.

Using Macro PDF417, large files are split into several file segments and encoded into individual symbols. The Control Block defines the file ID, the concatenation sequence and optionally other information about the file. The Macro PDF417 decoder uses the Control Block's information to reconstruct the file correctly, independent of symbol scanning order.

H.2 Macro PDF417 syntax

Each Macro PDF417 symbol shall encode a Macro PDF417 Control Block containing control information. The Control Block begins with the Macro marker codeword (928). The Control Block follows the data block with which it is associated, and the number of codewords in the control block is counted as data and incorporated in the value of the Symbol Length Descriptor. The beginning of the error correction codewords identifies the end of the Control Block.

NOTE A symbol containing no user data, other than a Macro PDF417 Control Block, is a valid symbol.

The Control Block shall contain at least the two mandatory fields: a segment index and file ID. It also may contain a number of optional fields, as described in H.2.3.

Figure H.1 illustrates the position of the Control Block in a Macro PDF417 symbol.

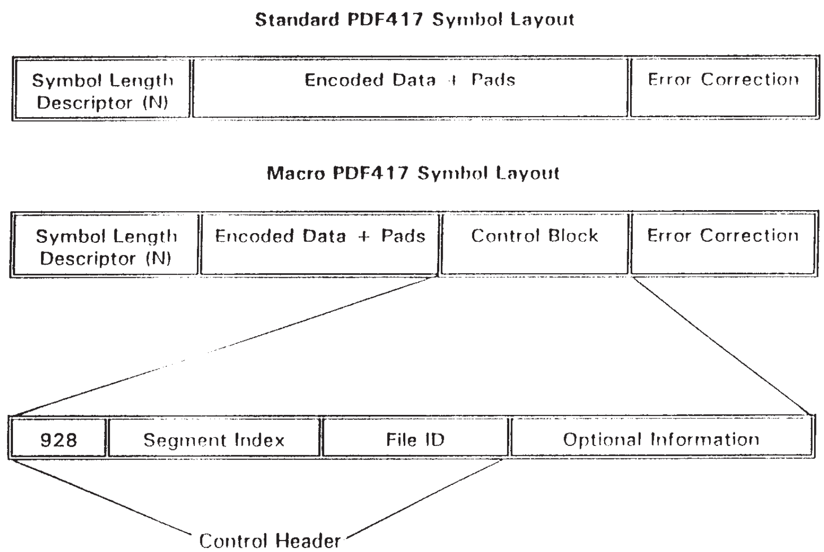


Figure H.1 — PDF417 Symbol Layouts

H.2.1 The segment index

In Macro PDF417, each symbol represents a segment of the whole file. To reconstruct the whole file, the segments need to be placed in the correct order. Control information in the Control Block facilitates this reassembly process. For a file divided into a set of j Macro PDF417 symbols, the segment index field in each symbol's Control Block contains a value between 0 and *j* - 1, corresponding to the relative position of that symbol's content within the distributed representation.

The segment index field is two codewords in length and is encoded using Numeric Compaction mode as defined in 5.4.4. The segment index value shall be padded with leading zeros to five digits before Numeric Compaction shall be applied, and the switch to Numeric Compaction shall not require an explicit mode latch (codeword 902). The largest allowed value in the segment index field is 99 998. Thus, up to 99 999 Macro PDF417 symbols may comprise the distributed representation of a data file.

NOTE This translates to a capacity of nearly 110 million bytes of data in Byte Compaction mode, or 184 million characters in Text Compaction mode, or nearly 300 million characters in Numeric Compaction mode.

H.2.2 File ID field

For each related Macro PDF417 symbol, the file ID field contains the same value. This ensures that all re-assembled symbol data belongs to the same distributed file representation. The file ID is a variable length field which begins with the first codeword following the segment index and extends to the start of the optional fields (if present) or to the end of the Control Block (if not).

Each codeword in the file ID can have a value between 0 and 899, effectively making the file ID a series of base 900 numbers. Each codeword of the series is transmitted as the 3-digit ASCII representation of its decimal value.

NOTE The effectiveness of the file identification scheme is influenced by both the length of the file ID field and the suitability of the algorithm used to generate its value.

H.2.3 Optional fields

Optional fields may follow the file ID. Each optional field begins with a specific tag sequence and extends until the start of the next optional field (if present) or the end of the Control Block (if not). The tag sequence consists of codeword 923 followed by a single codeword field designator. In each optional field, data following the tag sequence has a field-specific interpretation. Empty optional fields shall not be used. Table H.1 shows the correspondence between currently defined field designators and optional field contents. Each optional field begins with an implied reset to the compaction mode shown in the table and with an implied reset to ECI 000002 (or GLI 0 for encoders complying with earlier PDF417 standards). ECI escape sequences and mode latches and shifts may be used, but only in the optional fields initially in Text Compaction mode.

These fields shall always represent global file attributes and so need not be present in the Control Block of more than one Macro PDF417 symbol within the distributed file representation, with the exception of the segment count field, as described below. The segment which contains these fields is defined by the specific encoder implementation. If a particular field is to appear in more than one segment, it shall appear identically in every segment. There is no required order for the optional fields.

Table H.1 — Macro PDF417 Optional Field Designators

Field Designator	Byte Value Transmitted	Contents	Initial Compaction Mode	Fixed Compaction Mode[a]	Total Number of Codewords[b]
0	48	File Name	Text Compaction	N	Variable
1	49	Segment Count	Numeric Compaction	Y	4
2	50	Time Stamp	Numeric Compaction	Y	6
3	51	Sender	Text Compaction	N	Variable
4	52	Addressee	Text Compaction	N	Variable
5	53	File Size	Numeric Compaction	Y	Variable
6	54	Checksum	Numeric Compaction	Y	4

[a] A 'Y' in the 'Fixed Compaction Mode' column means that no ECIs and no compaction mode latches and shifts are allowed in that field.

[b] The totals shown in the last column include the two-codeword tag sequence.

As shown in Table H.1, all optional fields use standard PDF417 high-level encoding. At the beginning of each field, the default mode in effect shall be defined by Table H.1, regardless of mode shifts and latches earlier in the symbol.

Specific construction of optional fields shall be as follows.

— The segment count field (identifying the total number of Macro PDF417 symbols in the distributed file) can contain values from 1 to 99 999 and shall be encoded as two codewords. If the optional segment count field is used, that field shall appear in every segment.

— The time stamp field shall be interpreted in Numeric Compaction mode. It indicates the time stamp on the source file expressed as the elapsed time in seconds since 1970:01:01:00:00:00 GMT (i.e. 00:00:00 GMT on 1 January 1970). Using this format, four codewords can encode any date over the next 200 centuries.

— The file size field contains the size in bytes of the entire source file.

— The checksum field contains the value of the 16-bit (2 bytes) CRC checksum using the CCITT-16 polynomial $x^{16} + x^{12} + x^5 + 1$ computed over the entire source file.

The file size and checksum shall be calculated from the original source file, prior to the addition of any ECI escape sequences for Extended Channel Interpretation encoding. This implies that, if the receiver is to verify the checksum after reception, the original source file must be reconstructed verbatim. This requires, for the purposes of this optional checksum verification only, that no user-selectable or optional transformations of the byte stream be performed, even if these would normally be done in ECI decode processing.

If the CRC is used, the calculation may be performed either before the data is sent to the printer or in the printer, based on the capabilities of the printer.

Field designator values greater than 6 are not currently defined. However, PDF417 decoding equipment shall decode and transmit any optional fields encountered with a field designator of 7 to 9 (byte 55 to 57) or A to Z (byte 65 to 90) by treating the field's data as being initially in Text Compaction mode and being variable length.

H.2.4 Macro PDF417 terminator

The Control Block in the symbol representing the last segment of a Macro PDF417 file contains a special marker, consisting of the codeword 922 at the end of the Control Block. The Control Block for every other symbol shall end after any optional fields with no special terminator.

H.3 High level encoding considerations

While Macro PDF417 provides a mechanism for logically associating a set of symbols, it is important to realise that, with respect to PDF417 high-level encoding, each symbol shall remain a distinct entity. Thus, the scope of a mode switch shall be confined to the symbol in which it occurs. Each symbol shall implicitly begin in the Alpha sub-mode of the Text Compaction mode.

The two mandatory fields are encoded as follows: a) the segment index is encoded in Numeric Compaction mode; b) the file ID is encoded as a sequence of base 900 numbers.

In the context of a Control Block optional field, the compaction modes indicated in Table H.1 shall supersede the mode currently set by the mode identifier codewords within the data codeword region of the symbol. The scope of the current ECI, however, skips over the Macro Control Block to the start of the next Macro PDF417 symbol. Each Macro Control Block field begins with an implied reset to ECI 000002 (or GLI 0 for encoders complying with the earlier PDF417 standards). It shall also be possible to set a different ECI within an optional Text Compaction mode Macro Control Block field, for example, to represent properly a Greek addressee's name. The ECI escape sequence may be placed in any permitted position (see 5.5.3) after the tag codeword (923).

H.4 Encodation example

To illustrate the encodation of a Macro Control Block, the following example is used.

A Macro PDF417 series encodes a total of 4 567 bytes of user defined data in four PDF417 symbols (or file segments). Other 'header' data to be encoded are

— File ID = $17_{\text{base 900}}$ $53_{\text{base 900}}$,

— Segment count to be used,

— Sender: CEN BE, and

— Addressee: ISO CH.

NOTE The segment count, sender and addressee are three optional fields selected by the user.

On the assumption that the encoder places optional fields in the first symbol, the encodation of the Macro Control Block would be as follows for that symbol.

... [last data codeword] $[928]_A$ [111] $[100]_B$ [017] $[053]_C$ [923] $[001]_D$

[111] $[104]_E$ [923] $[003]_F$ [064] [416] $[034]_G$ [923] $[004]_H$ [258] [446] $[067]_I$

[first error correcting codeword]...

The last symbol of four would have the following Macro Control Block:

[last data codeword] $[928]_A$ [111] $[103]_B$ [017] $[053]_C$

923] $[001]_D$ [111] $[104]_E$ $[922]_J$ [first error correcting codeword]

where

A is the Macro Marker Codeword;

B is the File Segment ID.

File segments are numbered from 0 to *j* - 1, and are encoded using Numeric Compaction

1st Segment = 00000 = codewords 111, 100

4th Segment = 00003 = codewords 111, 103

C = File ID to base 900

D = Tag for segment count field

E = Segment count

F = Tag for sender field

G = Sender field encoding CEN BE

H = Tag for addressee field

I = Addressee field encoding ISO CH

J = Macro PDF417 Terminator

H.5 Macro PDF417 and the Extended Channel Interpretation protocol

The symbology-independent Extended Channel Interpretation (ECI) protocol was developed after PDF417 was specified as a symbology. PDF417 supported its own Global Label Identifier (GLI) system, the precursor and basis of the ECI protocol, from the first publication of the symbology specification in 1994. Therefore, previous 'GLI' implementations have to be taken into account. There are two different conditions which need to be taken into account:

— GLI 0 and 1 which were the only interpretations specified in the original PDF417 specifications. These are equivalent to ECI 000000 and ECI 000001. The precise rules for Macro PDF417 are defined in H.5.1;

— all other ECI assignments, whose usage with Macro PDF417 is defined in H.5.2.

H.5.1 Macro PDF417 with ECI 000000 and 000001 (GLI 0 and 1)

As GLIs were intrinsically part of the original PDF417 specification, it is logical to have a GLI encoder and Macro PDF417 encoder combined in one unit. The original PDF417 symbology specification called for an implied 'return-to-GLI 0' logic at the beginning of the second and subsequent Macro PDF417 symbols, thus every symbol is expected to start at the default interpretation. For GLI 0 and 1 (equivalent to ECI 000000 and ECI 000001), this has no inherent effect on the encodation. However, for some complex ECIs, the return-to-GLI 0 logic is difficult to implement in a symbology-independent manner.

Encoding software compliant with the original specification for Macro PDF417 and GLI 0 and 1 is completely suitable for pre-existing applications. So too are pre-existing applications of user defined GLIs (now called ECIs) because by definition, the domain of the system is constrained.

All ECIs numbered 000002 or higher shall not be defined with the return-to-GLI 0 logic. Therefore, PDF417 symbols shall not mix ECI 000000 and ECI 000001 with any higher numbered ECI (except in closed systems).

H.5.2 Macro PDF417 and other ECIs

An ECI encoder could be symbology independent and create a byte stream as input to a PDF417 symbology encoder. The ECI encoder should behave as if there is a single data stream, irrespective of the size of the file. Thus, an ECI once invoked would persist across segments until another ECI or the end of the encoded data. This is essential if, for example, the ECI assignment represents an encryption scheme, where returning to GLI 0 would not be appropriate.

Macro PDF417 encoders compliant with this International Standard need not encode the prevailing ECI at the beginning of subsequent Macro PDF417 symbols.

NOTE There may need to be some iteration to produce a logical end-of-symbol encodation, for example, Numeric Compaction mode shall not straddle two segments, but two separate Numeric Compaction blocks can be encoded at the end of one symbol and at the beginning of the next. These conditions are related to Macro PDF417 and High Level Encoding (see H.3) and not Macro PDF417 and ECIs.

H.6 Macro PDF417 data transmission

The transmission of Macro PDF417 Control Block information shall be treated in a similar manner to that of interpretative ECIs. The symbology-independent ECI protocol is defined below; the original PDF417 protocol is defined in Annex M. Although the Macro Control Block is encoded at the end of the symbol's data, it is transmitted before the symbol's data when using the ECI protocol.

Three codewords (922, 923 and 928) signal the encodation of a Macro PDF417 Control Block or one of its constituent parts. Decoding is as follows:

a) If the Macro marker codeword (928) begins the sequence:

   1) Codeword 928 is transmitted as the escape sequence 92, 77, 73, which represents '\MI' in the default interpretation.

   2) The next two codewords identify the segment index. These are encoded in Numeric Compaction mode and decode as a 5-digit number in the range 00 000 to 99 998.

   3) The next codewords encode the file ID field, which shall be the same for all related Macro PDF417 symbols. The end point of the file ID field is codeword 922, codeword 923, or the end of the encoded data in the symbol. Each codeword is converted to a 3-digit number in the range 000 to 899 (i.e. the codeword number) and transmitted as three byte values (in the range decimal 48 to 57) following the escape header 92, 77, 70, which represents '\MF' in the default interpretation.

b) If the Macro sequence tag codeword (923) begins the sequence:

   1) Codeword 923 is transmitted as the escape sequence 92, 77, 79, which represents '\MO' in the default interpretation.

   2) The next codeword represents one of the optional field designators in Table H.1 transmitted as a single byte representing the ASCII value of the designator.

   3) The next codewords carry the data content of the optional field designator. The end point of the optional field is codeword 922, codeword 923, or the end of the encoded data in the symbol. The intervening codewords should be converted according to the decode rules of the relevant compaction mode defined in Table H.1. The resultant data may be variable length.

c) If the Macro PDF417 Terminator (codeword 922) is identified, the escape sequence 92, 77, 90, which represents '\MZ' in the default interpretation, shall be transmitted.

d) At the end of the Macro Control Block, as defined by the end of encoded data in the symbol, the escape sequence 92, 77, 89, which represents '\MY' in the default interpretation, shall be transmitted.

NOTE This escape sequence is not explicitly encoded in the symbol.

All the Macro Control Block fields for a symbol (segment) shall be transmitted as a single block starting with \MI... and ending with \MY. The transmission of the Macro Control Block shall precede the transmission of the remainder of the encoded file segment, even though it is encoded at the end of the symbol.

EXAMPLE

The Macro PDF417 Control Block of the first symbol, Segment Index = 0, with a File ID (100, 200, 300) would be encoded in the symbol as the codeword sequence:

[928] [111] [100] [100] [200] [300]

It would be transmitted as:

Data transmission (byte):

92, 77, 73, 48, 48, 48, 48, 48, 92, 77, 70, 49, 48, 48, 50, 48, 48, 51, 48, 48, 92, 77, 89

ASCII interpretation:

\MI00000\MF100200300\MY

As the Macro PDF417 symbols are scanned, the de-packetizing function reconstructs the original message, bearing in mind that the symbols may be scanned out of sequence. If the system is operating in buffered mode, the de-packetizing function is in the decoder; if operating in unbuffered mode, it is in the receiving system.

Decoders should provide a decoder-specific means whereby the processing of a given Macro PDF417 file ID may be aborted, thus allowing the decoder to begin processing a new File ID. This is necessary to prevent a deadlock condition should one or more symbols of a given File ID be missing or undecodable.

H.6.1 Operating in buffered mode

In buffered mode, de-packetizing shall be performed in the decoder/reader. Depending on the equipment configuration, it will either

— send the reconstructed data with no Macro Control Block, or

— send one Macro Control Block (which itself may have been reconstructed to include all optional fields included in any symbols) to precede the entire encoded message. The resulting Macro Control Block shall have its Macro Index field set to 0 and shall include the Macro end-of-file field (in effect, to mark the entire reconstructed message as the first - and only - Macro segment of the pseudo-series).

H.6.2 Operating in unbuffered mode

In unbuffered mode, de-packetizing shall be performed in the receiving system. Each transmitted Macro Control Block shall represent all of the required and optional fields actually encoded in the symbol.

When configured in unbuffered mode, a decoder may optionally be configured not to require successive symbols to be of the same File ID. This procedure would only be appropriate if the decoder is configured to transmit the Macro PDF417 Control Block to the receiving system, and this receiving system is designed to monitor the File ID portion of the Control Block to determine when the entire file has been processed. Symbols with a different File ID or no File ID (e.g. a single symbol not part of a Macro PDF417 set) shall be dealt with as determined by the receiving system.

To facilitate checking that all symbols in a Macro PDF417 set are received in an unbuffered operation, the optional Segment Count field should be used whenever possible as part of the encoded Macro Control Block.

H.6.3 Reset-to-Zero transmissions

Because the original AIM USA (1994) and AIM Europe (1994) PDF417 specifications defined GLI 0 and GLI 1 to have rules slightly different from the rules for ECIs, a reader compliant with this International

Standard must, in two situations, emit extra escape sequences when transmitting symbols containing explicit GLI 1 invocations.

a) The decoder shall transmit either a GLI 0 escape sequence or an ECI 000000 escape sequence (depending upon which transmission protocol it is programmed to use) after transmitting the data of any Macro PDF417 symbol whose data ends in a GLI 1 (ECI 000001) interpretation.

b) The decoder shall transmit a GLI 1 (ECI 000001) at the start of each variable length optional field encoded in Text Compaction mode in the Macro Control Block, if the data preceding that field ends in a GLI 1 (ECI 000001) interpretation.

This requirement applies whether operating in buffered or unbuffered mode, and whether the decoder is programmed to transmit using either the ECI protocol, or the original PDF417 transmission protocol.

Annex I
(normative)

Testing PDF417 symbol quality

As specified in 5.14.4, the quality of PDF417 symbols is evaluated according to the methodology defined in ISO/IEC 15415 for the assessment of multi-row symbologies with cross-row scanning ability.

In summary, PDF417 symbols are graded in respect of the following:

— analysis of the scan reflectance profile, applied to the start and stop patterns only;

— Codeword Yield, applied to the data and error correction codewords only, which measures the efficiency with which linear scans can recover data from the symbol. The Codeword Yield is the number of validly decoded codewords expressed as a percentage of the maximum number of codewords that could have been decoded, i.e. the number of data columns in the symbol multiplied by the number of “qualified” scans (after adjusting for tilt);

— Unused Error Correction, applied to the data and error correction codewords only, which expresses the number of errors and erasures as a function of the error correction capacity of the symbol;

— codeword print quality, applied to the data and error correction codewords only, which enables the Decodability, Defects and Modulation parameters of scan reflectance profiles covering the entire data region of the symbol to be graded; these grades are then modified to allow for the effect of error correction in masking less than perfect attributes of the symbol that influence symbol quality.

The overall symbol grade shall be the lowest of the grade based on analysis of the scan reflectance profile, and the grades based on Codeword Yield, Unused Error Correction and codeword print quality.

Annex J
(normative)

Reference decode algorithm for PDF417

J.1 General

This Annex describes the reference decode algorithm used in the computation of decodability when assessing the symbol quality using the method described in ISO/IEC 15415.

When assessing symbol quality through the use of this reference decode algorithm, a PDF417 symbol shall be decoded, in a series of scan lines running across the symbol that cross at least one start or stop character, but not necessarily row by row. It is possible to decode the symbol if the scan line crosses two or more rows by using the cluster number. The decoding of symbol character bar-space sequences shall be achieved by using 'edge to similar edge' (e) measurements.

The PDF417 symbol shall be decoded in four phases:

a) initialisation - to establish the symbol matrix;

b) line decoding using the reference decode algorithm;

c) filling the matrix;

d) interpretation.

J.2 Initialisation

A sufficient number of line decodes (see J.3) shall be performed at the start of the decode process to establish the symbol structure parameters [number of rows (r), number of columns (c)], and error correction levels. This information is encoded in the left and right row indicators, adjacent respectively to the start and stop characters.

After the symbol structure parameters have been initialised, a matrix shall be established which reflects the size (rows by columns) of the symbol being decoded. The matrix shall exclude start and stop characters and row indicators.

J.3 Reference decode algorithm for line decoding

A decodable scan line shall contain at least one quiet zone, a start or stop character, one row indicator and one or more symbol characters in the data region. A scan line may cross more than one row. The algorithm contains the following steps to decode the line.

a) Confirm the presence of a quiet zone.

b) For each symbol character bar-space sequence (including start and stop character), calculate the following width measurements as per Figure J.1.

p

e_1, e_2, e_3, e_4, e_5 and e_6

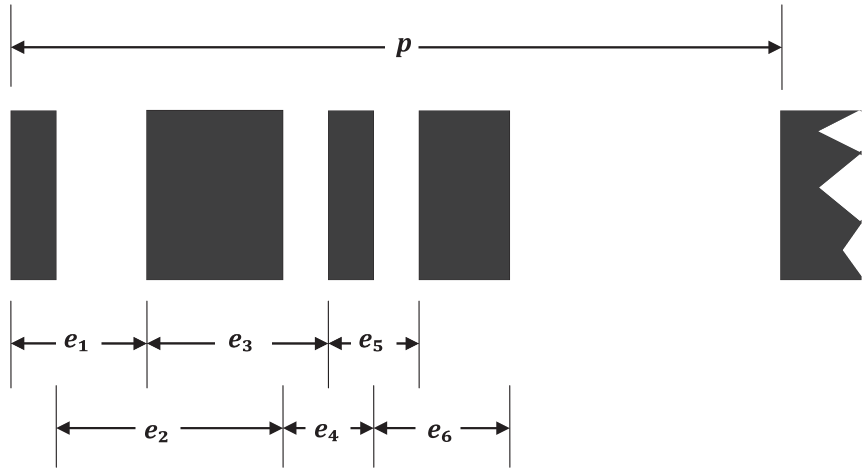


Figure J.1 — Decode measurements

c) Convert measurements e_1, e_2, e_3, e_4, e_5, and e_6 to normalised values E_1, E_2, E_3, E_4, E_5 and E_6 which will represent the integral module width of these measurements. The following method is used for the i th value.

If $1{,}5p/17 \le e_i < 2{,}5p/17$, then $E_i = 2$

If $2{,}5p/17 \le e_i < 3{,}5p/17$, then $E_i = 3$

If $3{,}5p/17 \le e_i < 4{,}5p/17$, then $E_i = 4$

If $4{,}5p/17 \le e_i < 5{,}5p/17$, then $E_i = 5$

If $5{,}5p/17 \le e_i < 6{,}5p/17$, then $E_i = 6$

If $6{,}5p/17 \le e_i < 7{,}5p/17$, then $E_i = 7$

If $7{,}5p/17 \le e_i < 8{,}5p/17$, then $E_i = 8$

If $8{,}5p/17 \le e_i < 9{,}5p/17$, then $E_i = 9$

Otherwise, the symbol character bar-space sequence is in error.

d) After finding a start or stop character, attempt to decode a row indicator, and as many symbol characters as the number of columns in the matrix, in the direction derived from the start or stop character decoded. Decode the symbol character bar-space sequences as per step 5.

e) Compute the symbol character cluster number K by:

$K = (E_1 - E_2 + E_5 - E_6 + 9) \bmod 9$

NOTE 1 This formula yields identical results to the equation given in 5.3.1.

The cluster number K shall equal 0, 3 or 6; otherwise the symbol character and its associated codeword are in error.

f) Retrieve the codeword from the decode table (Annex A) using the seven values (cluster value K and the values E_1, E_2, E_3, E_4, E_5 and E_6) as the key. These values can be calculated directly from the bar-space sequence values given in Annex A.

NOTE 2 The calculation implicitly uses the cluster number to detect all decode errors caused by single non-systematic one-module edge errors.

g) Once valid start and/or stop characters have been established, the codewords for the left row indicator and/or right row indicator shall be used to establish the symbol structure parameters. The inverse of the equations defined in 5.11.3.1 and 5.11.3.2 shall be used to establish the row number (*F*), the number of rows (*r*), the number of columns (*c*) and the error correction level (*s*).

h) Perform such other secondary checks (scan acceleration, absolute timing dimensions, quiet zones etc) as deemed prudent and appropriate for the particular characteristics of the reading device.

J.4 Filling the matrix

The following procedure shall be used to fill the matrix of rows (*r*) by columns (*c*) established by the initialisation procedure.

a) Set the initial value of the erasure count *v* to be equal to *r* x *c*.

b) For each scan, attempt to decode as many codewords as the number of columns of the matrix.

c) Valid decode results are placed in the matrix at their appropriate positions determined by the row number (from the row indicators) and the cluster value.

If row crossing occurs, the scan line will have different row numbers indicated by the left and right row indicators. The cluster number shall be used to interpolate the correct row number for each individual valid codeword.

EXAMPLE A decoded scan has valid start and stop characters and has a left row indicator with row number 7 and a right row indicator with row number 10. There are 10 columns in the matrix. The scan line has not decoded three codewords because it did not remain entirely in the one row for the full transition, however the position of these ‘missing’ codewords is known from element timings.

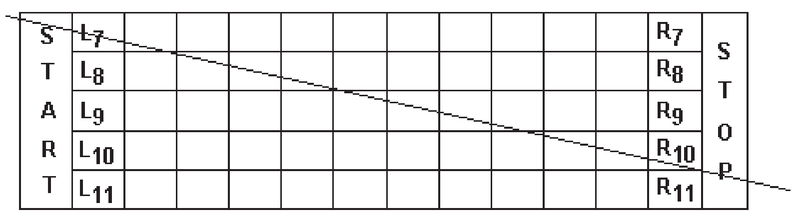


Figure J.2 — Schematic Showing a Scan Line Crossing Rows

The clusters are as follows: unknown, 6, 6, 6, unknown, 0, 0, unknown, 3, 3.

Using matrix notation of *r* (row), *c* (column), the codewords are filled in the positions:

unknown, (8, 2), (8, 3), (8, 4), unknown, (9, 6), (9, 7), unknown, (10, 9), and (10, 10)

NOTE This example is extreme in that it crosses four rows, but it still results in the successful decode of 70 percent of the codewords.

d) As the matrix is being filled, the erasure count *v* shall be reduced by one for each valid codeword.

e) If the error correction level is not equal to zero, error recovery may be attempted when the number of unknown codewords (the erasure count *v*) satisfies the equations in 5.7.2 (with *v* = *l* and *f* = 0). If error recovery fails, then more codewords shall be collected.

f) If the error correction level is equal to zero, validate the two error correction codewords.

For more details on error detection and correction see Annex K.

J.5 Interpretation

Beginning from an initial state of the Alpha sub-mode of Text Compaction mode, the data codewords shall be interpreted according to the compaction modes.

Annex K
(normative)

Error correction procedures

When the total number of unknown codewords **v** is less than or equal to the value of l in the appropriate equation in 5.7.2, where $f = 0$, then the recovery scheme may be invoked. The unknown codewords shall be substituted by zeros and the position of the l th unknown codeword is j_l for $l = 1,2,..., v$. Construct the symbol character polynomial:

$$C(x) = C_{n-1}x_{n-1} + C_{n-2}x^{n-2} + ... + C_1x^1 + C_0$$

where

n coefficients are the codewords read, with C_{n-1} being the first codeword;

n is the total number of codewords.

Calculate k syndrome values (S_1 to S_k) by evaluating:

$C(x)$ at $x = 3^i$

for $i = 1$ to $i = k$

where k is the number of error correction characters in the symbol = 2^{s+1}.

A circuit to generate the syndromes is shown in Figure K.1.

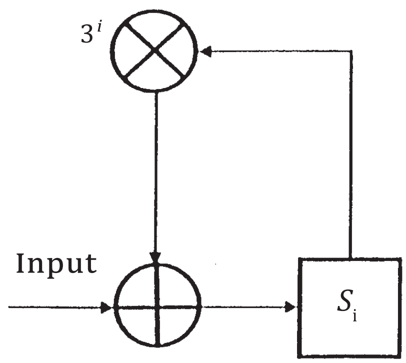


Figure K.1 — Symbol Syndrome Divider

Since the locations of unknown codewords in the symbol matrix are known from j_l for $l = 1, 2, ... v$, the error location polynomial for these known positions can be computed:

$$\sigma(x) = (1 - \beta_1 x)(1 - \beta_2 x)...(1 - \beta_v x)$$

$$= 1 + \sigma_1 x + ... + \sigma_v x^v$$

where $\beta_l = 3^{j_l}$.

The error location polynomial, σ(x), can be updated to include the position of errors. This can be done by using the Berlekamp-Massey algorithm, see Reference [2].

At this point, verify that the number of erasures and errors satisfy the appropriate error correction capacity equation in 5.7.2.

Solving σ(x) = 0 yields the position of the t errors, where $t \geq 0$; if $t = 0$ there is no error. It is now necessary to compute the error value, e_{jl} for location j_l, $l = 1, ..., v + t$. To compute the error values one auxiliary polynomial, the Z-polynomial, is needed which is defined by:

$$Z(x) = 1 + (s_1 + \sigma_1)x + (s_2 + \sigma_1 s_1 + \sigma_2)x^2 + \ldots + (s_\eta + \sigma_1 s_{\eta-1} + \sigma_2 s_{\eta-2} + \sigma_\eta)x^\eta$$

where $\eta = v + t$.

The error value at location j_l is thus given by:

$$e_{j_l} = \frac{Z(\beta_l^{-1})}{\beta_l \prod_{i=1, i \neq l}^{\eta} (1 - \beta_i \beta_l^{-1})}$$

After solving successfully for the error values, the complements of the error values are added to the codewords in the corresponding locations.

Annex L
(normative)

Symbology identifier

The uniform methodology defined in ISO/IEC 15424 shall be used for reporting the symbology read, options set in the reader and any special features of the symbology encountered.

The symbology identifier for PDF417 is:

]Lm

where

] is the symbology identifier flag character (ASCII 93);

L is the symbology identifier for PDF417;

m is a modifier character with one of the values defined in Table L.1.

Table L.1 — Symbology Identifier Modifier Values for PDF417

m	Option
0	Reader set to conform with protocol defined in the original AIM USA (1994) and AIM Europe (1994) PDF417 specifications (see Annex M)[a]
1	Reader set to follow the protocol of this standard for Extended Channel Interpretation (see 5.17.2). All data characters 92 are doubled
2	Reader set to follow the protocol of this standard for Basic Channel operation (see 5.17.1). Data characters 92 are not doubled[b]

[a] When this option is transmitted, the receiver cannot determine reliably whether ECIs have been invoked, nor whether data byte 92 has been doubled in transmission.

[b] When decoders are set to this mode, unbuffered Macro PDF417 symbols, and symbols requiring the decoder to convey ECI escape sequences, cannot be transmitted.

This information shall not be encoded in the bar code symbol, but should be generated by the decoder after decoding and be transmitted as a preamble to the data message.

Annex M
(normative)

Transmission protocol for decoders conforming with original PDF417 standards

M.1 General

Earlier PDF417 symbology specifications supported: Basic Channel Mode, Global Label Identifiers (the precursor to the symbology-independent Extended Channel Interpretation) and Macro PDF417 (but without full integration with the ECI protocol). This Annex

— defines the transmission protocol compliant with the original specification, and which may still be used, and

— addresses issues of compatibility.

M.2 Basic Channel mode

In the Basic Channel, all data symbol characters are translated according to the compaction modes in effect, and are included in the data transmission as a sequence of 8-bit bytes. Start and stop characters, row indicators, the Symbol Length Descriptor, mode switching codewords and error correction codewords are not transmitted.

NOTE This is identical to the procedure of 5.17.1.

Original decoders should output symbology identifier]L0, or may not transmit a symbology identifier preamble.

M.3 GLI encoded symbols

Only GLI 0 and GLI 1 have been previously specified, but the transmission of all GLI/ECI escape sequences is supported by the original protocol. Three codewords (925, 926 and 927) signal the encodation of a GLI value and are decoded as byte values as follows:

a) If the GLI sequence begins with codeword 927:

   1) Codeword 927 is transmitted as a 4-byte escape sequence 92, 57, 50, 55, which represents '\927' in the ASCII interpretation.

   2) The next codeword represents the GLI number in the range 000 to 899. The codeword is converted to a 3-digit value. The 3-digit value is transmitted as the appropriate byte values (48 to 57), preceded by byte 92.

   EXAMPLE

   Symbol encodes: [927] [001]

   Data transmission (byte): 92, 57, 50, 55, 92, 48, 48, 49

   ASCII interpretation: \927\001

b) If the GLI sequence begins with codeword 926:

1) Codeword 926 is transmitted as a 4-byte escape sequence 92, 57, 50, 54, which represents '\926' in the ASCII interpretation.

2) The next two codewords (codewords 000 to 899 are permissible) represent the number of the ECI as follows:

Codeword 1 = ECI_no div 900 - 1

Codeword 2 = ECI_no mod 900

Each codeword is converted to a 3-digit value. The 3-digit value is transmitted as the appropriate byte values (48 to 57), preceded by byte 92.

EXAMPLE

Symbol encodes: [926] [136] [156]

Data transmission (byte): 92, 57, 50, 54, 92, 49, 51, 54, 92, 49, 53, 54

ASCII interpretation: \926\136\156

c) If the GLI sequence begins with codeword 925:

1) Codeword 925 is transmitted as a 4-byte escape sequence 92, 57, 50, 53, which represents '\925' in the ASCII interpretation.

2) The next codeword represents the number of the user defined GLI minus 810 900 (any codeword 000 to 899 is permissible). This codeword is converted to a 3-digit value. The 3-digit value is transmitted as the appropriate byte values (48 to 57), preceded by byte 92.

EXAMPLE

Symbol encodes: [925] [456]

Data transmission (byte): 92, 57, 50, 53, 92, 52, 53, 54

ASCII interpretation: \925\456

The procedure is repeated for each occurrence of a GLI.

NOTE 1 Illustrations of similar ECI examples, but utilizing the ECI protocol, are given in 5.17.2.

If the reverse solidus, or other character represented by byte 92 needs to be used as encoded data, transmission shall be as follows. Whenever byte 92 occurs as data, two bytes of that value shall be transmitted; thus a single occurrence is always an escape character and a double occurrence indicates true data.

EXAMPLE

Encoded data: A\\B\C

Transmission: A\\\\B\\C

The default escape character may be changed in the decoder (in which case the receiving system shall be configured to match), but the byte values 47 to 58 (generally interpreted as numeric digits) shall not be used.

NOTE 2 In the ECI compliant protocol (see 5.17.2), the escape character is fixed at 92.

As an option, decoders may have an operating mode where no escape character is defined; such readers cannot transmit escape sequences nor double any data characters. Thus, this mode cannot support the transmission of ECI escape sequences, nor Macro PDF417 Control Blocks.

M.4 Macro PDF417 symbols

When operating under the original PDF417 transmission protocol, once a PDF417 decoder has processed a Macro PDF417 symbol with a given file ID, it shall decode and transmit all of the symbols for that file ID before it may transmit any other symbols. This requirement applies under either of the following transmission modes.

M.4.1 Transmission in buffered mode

A buffered transmission system requires the decoder to collect the entire symbol set prior to its transmission. Processing of the mandatory fields of the Macro Control Block is dealt with internally. The transmission of optional fields can be individually enabled or disabled in the decoder. Optional fields, if present, should be transmitted once at the end of the entire data set. Each field shall begin with the transmission of the corresponding Macro PDF417 optional field tag sequence. The tag sequence consists of the codeword 923 followed by a tag value as defined in Table H.1; this sequence shall be transmitted using the escape character as defined in M.3. The high level decoded content of the field shall be transmitted after this tag sequence.

M.4.2 Transmission in unbuffered mode

An unbuffered transmission system allows the decoder to transmit the individual symbols as they are decoded.

When using the unbuffered scheme, transmission of the Macro PDF417 Control Header should be enabled, because symbols in the unbuffered scheme are not ordered internally by the reader. This allows the host system to impose the proper ordering on the received data.

Transmission of the Macro PDF417 Control Header may be enabled or disabled. The Macro PDF417 Control Header is a portion of the Macro PDF417 Control Block, (see Figure H.1) which consists of the marker codeword 928, the Segment Index (in Numeric Compaction mode), and the File ID codeword sequence. When transmission of the Control Header is enabled, the marker codeword and the File ID codewords should be transmitted using the escape character as defined in M.3. For example, the Macro PDF417 Control Header of the first symbol, Segment Index = 0, with a File ID (100, 200, 300) would be encoded in the symbol as the codeword sequence:

[928] [111] [100] [100] [200] [300]

and (assuming the default escape character 92) would be transmitted as:

Data transmission (byte):

92, 57, 50, 56, 48, 48, 48, 48, 48, 92, 49, 48, 48, 92, 50, 48, 48, 92, 51, 48, 48

ASCII interpretation: \92800000\100\200\300

If enabled, the Macro PDF417 Control Header shall be transmitted following the data encoded in the symbol.

When the last GLI sequence transmitted by the reader is other than GLI 0, then the transmitted data from that segment shall be terminated with the byte sequence 92, 57, 50, 55, 92, 48, 48, 48 (ASCII equivalent: \927\000), as if the symbol's data ended with the sequence of codewords [927] [000]. This reverts the interpretation of the next block back to GLI 0.

The transmission of optional fields can be individually enabled or disabled in the decoder. The enabled optional fields shall be transmitted with each Macro PDF417 symbol in which they have been encoded. Each field shall begin with the transmission of the corresponding Macro PDF417 optional field tag sequence. The tag sequence consists of the codeword 923 followed by a tag value as defined in Table H.1; this sequence shall be transmitted using the escape character as defined in M.3. The high level decoded content of the field shall be transmitted after this tag sequence.

Based only on the transmission of the encoded data stream, it can be difficult or impossible to determine where the boundary exists between the end of the Macro Control Block (especially if containing optional fields) and the beginning of the next symbol's data content. The system's transmission protocol (e.g. using the conventional transmission escape characters STX and ETX, or other 'hand shaking' procedures) may be used to determine the boundaries between transmitted Macro PDF417 symbols.

To facilitate checking that all symbols in a Macro PDF417 set are received in an unbuffered operation, the optional Segment Count field should be used whenever possible as part of the encoded Macro Control Block.

M.5 Transmission of reserved codewords using the original PDF417 protocol

When operating under the original PDF417 transmission protocol, decoders should transmit a reserved codeword as an escape character (default of 92) followed by three digits which represent the decimal value of the reserved codeword. The data codewords which follow the reserved codeword are interpreted and transmitted according to the compaction mode in effect prior to the reserved codeword. Specifically, the interpretation will be as if the reserved codeword inserted a latch codeword to the compaction mode already in effect.

Such a latch, when in Byte or Numeric Compaction mode, re-initialises a new 'grouping' of codewords. If the prevailing mode is Text Compaction, the effect is to re-initialise to the Alpha sub-mode of Text Compaction.

While this protocol can properly transmit the message syntax of any reserved codeword whose future definition is to provide a signalling function, it will not provide unambiguous output for a new compaction mode. Therefore, when using the original PDF417 transmission protocol, the receiver should discard any data following the escape sequence representing a newly defined compaction mode codeword.

M.6 Achieving compatibility between old and new PDF417 equipment

M.6.1 Encoders

The introduction of Extended Channel Interpretations, which are symbology-independent, means that it is logical to separate the functions of ECI encoding from symbology encoding. GLI encoding is de facto intrinsically linked to the PDF417 symbology. The encoded codeword stream is intended to be equivalent, whether the symbol has been encoded on existing or new encoders. It should be possible to encode, for example, data conforming with the interpretation of ECI 000123 (which itself has not been defined at the date of publication of this standard) under a PDF417 specific GLI capable encoder or on a first stage symbology-independent ECI encoder followed by a second stage PDF417 symbology encoder.

There are two constraints:

— the return-to-GLI 0 logic shall only be applied to GLI 0 (ECI 000000) and GLI 1 (ECI 000001);

— GLI 0 and 1 shall not be intermixed with other ECIs in the same symbol, or Macro PDF417 set.

M.6.2 Decoders

The key to interoperability between original and new protocol PDF417 decoders is the required transmission of the symbology identifier prefix whenever a decoder is configured for the new Extended Channel Mode operation, and the required use of the prefix whenever old and new PDF417 equipment is mixed at the same installation. That is, a decoder enabled for Extended Channel Mode operation (even if reading a mix of Basic Channel Mode and Extended Channel Mode symbols) will send a symbology identifier with every transmission.

NOTE The original AIM USA (1994) and AIM Europe (1994) PDF417 specifications did not require the use of a symbology identifier, even when doubling the escape character (default of 92). Compliance with the ECI protocol as specified in this standard requires the use of the symbology identifier.

Decoders shall be considered to be in conformance under one of the following conditions.

a) **Fully conforming with the ECI protocol and with this International Standard**:

   1) Transmitting the appropriate symbology identifiers.

   2) Capable of being set or switched to Basic Channel Mode or Extended Channel Mode operation.

   3) Transmitting the ECI protocol as specified in this standard (see 5.17.2).

   4) Processing Macro PDF417 as specified in this standard.

b) **Conforming with 1994 standards**:

   1) And interoperable with new equipment, and ECI encoded symbols.

      i) Transmitting the symbology identifier ‘]L0’.

      ii) Capable of being set or switched to Basic Channel Mode or Extended Channel Mode operation.

      iii) Transmitting the GLI protocol as specified in M.3.

      iv) Processing Macro PDF417 as specified in M.4.

   2) But not interoperable with new equipment, and ECI encoded symbols.

      i) Not transmitting a symbology identifier.

      ii) Capable of being set or switched to Basic Channel Mode or Extended Channel Mode operation.

      iii) Transmitting the GLI protocol as specified in M.3.

      iv) Processing Macro PDF417 as specified in M.4.

c) **Conforming with Basic Channel Mode only**:

   1) Transmitting the symbology identifier ‘]L0’ (old equipment) or ‘]L2 (new equipment), or transmitting no symbology identifier.

   2) Treating symbols containing ECI codewords as invalid.

   3) Treating Macro PDF417 symbols as invalid, unless the reader is operating in buffered mode, and transmission of the Macro Control Header is disabled.

Assuming that equipment is properly set up as above, this gives the receiver the ability to detect, and react properly to, the following conditions.

a) **If a symbology identifier of ‘]L1’ is present at the start of the transmission**:

   In this case, the receiver can be sure that the decoder is operating in Extended Channel Mode for the symbol scanned. Therefore all byte 92, when occurring as data, have been doubled whether or not the symbol contains ECIs or is part of a Macro PDF417 set. Single occurrences of byte 92 indicate the start of an escape sequence. All other features conform with this standard.

b) **If a symbology identifier of ‘]L2’ is present at the start of the transmission**:

   In this case, the receiver can be sure that the decoder is operating in Basic Channel Mode for the symbol scanned. Therefore, byte 92 will always represent a single byte of data.

   Symbols with ECI escapes shall be considered invalid. Macro PDF417 symbols shall be considered invalid, unless the reader is configured for buffered mode, and is configured to not transmit Macro PDF417 Control Headers.

c) **If a symbology identifier of ']L0' is present at the start of the transmission, denoting the 1994 version of PDF417:**

This case is an exception because the original AIM USA (1994) and AIM Europe (1994) PDF417 specifications, although it does provide explicit Extended Channel Mode support, define '0' (i.e. 'no options set') as the only option value for the PDF417 symbology identifier. Thus, existing PDF417 equipment, if fully compliant with the original AIM USA (1994) and AIM Europe (1994) PDF417 specifications, will not use the new option values to indicate whether Extended Channel Mode or Basic Channel Mode is in effect. Therefore, if the receiver sees ']L0', then it should expect 1994 standard-compliant PDF417 behaviour. In particular:

1) The receiver cannot tell from the transmission whether the decoder is in Extended Channel Mode (always doubles the byte assigned as the escape character as per M.3) or Basic Channel Mode (never doubles any bytes); the decoder must be configured to match the expectations of the receiver.

2) If the decoder is set to Extended Channel Mode and if ECIs are encoded in the symbol, the decoder will transmit 1994 PDF417-style GLI escape sequences (as per M.3) rather than ECI escape sequence as defined in 5.17.2.

3) Using original protocol, if a Macro Control Block is present, the contents of the Macro Control Block follows, rather than precedes, the data bytes in the symbol.

d) **If no symbology identifier is present at the start of the transmission:**

In this case, either

1) the decoder is properly configured to support Basic Channel Mode symbols only. The receiving system is assured that no byte value is being doubled by the decoder and that any apparent ECIs in the data stream are accidental character combinations, or

2) the decoder is improperly configured for interoperability in an open system where ECI encoded symbols may be encountered.

Annex N
(informative)

Algorithm to minimise the number of codewords

The same data may be represented by different PDF417 codeword sequences through the use of different compaction modes and switching procedures. There shall be no prescribed procedure, but the following algorithm will tend to minimise the number of codewords required.

a) Let *P* point to the start of the data stream.

b) Set current encoding mode to Text Compaction.

c) Let *N* be the number of consecutive digits starting at *P*.

d) If *N* is ≥13 then

 1) latch to Numeric Compaction mode,

 2) encode the *N* characters using numeric compaction,

 3) advance *P* by *N*, and

 4) go to Step 3.

e) Else if *N* <13 then

 1) let *T* be the length of a Text Compaction mode character sequence starting at *P*. The sequence is terminated when either a character from a mode other than Text Compaction is detected or a numeric sequence of ≥13 digits is detected,

 2) if *T* is ≥5 then

  i) latch into Text Compaction mode,

  ii) encode the *T* characters using the Text Compaction mode,

  iii) advance *P* by *T*, and

  iv) go to Step 3,

 3) else if *T* <5 then

  i). let *B* be the length of the binary encodable sequence starting at *P*. The sequence is terminated when either a Text Compaction sequence of length ≥5 is found or a numeric sequence of length ≥13 is found,

  ii) if *B* is equal to 1 AND the current mode is Text Compaction, then

   I) shift into Byte Compaction mode,

   II) encode the single byte value using Byte Compaction mode,

   III) advance *P* by *B*, and

   IV) go to Step 3,

  iii) else

   I) latch into Byte Compaction mode,

II) encode the *B* bytes using Byte Compaction mode,

III) advance *P* by *B*, and

IV) go to Step 3.

Annex O
(informative)

Guidelines to determine the symbol matrix

O.1 Parameters affecting the determination of the matrix

A number of parameters should be used before printing to determine the symbol matrix in terms of the number of rows (r) and columns (c).

Each parameter addresses one single feature which may constrain the symbol matrix. In the equations which follow, A, c, k, n, Q_H, Q_V, r, X and Y conform with the definitions provided in 4.1.

The equations may be used in their own right or to construct a more complex algorithm.

— Parameter 1: Number of Rows: r

$3 \leq r \leq 90$ (see 5.2.1)

— Parameter 2: Number of Columns: c

$1 \leq c \leq 30$ (see 5.2.2)

— Parameter 3: X dimension

Defined by the application specification (see 5.8.1)

— Parameter 4: Y dimension

$Y \geq 3X$ (see 5.8.2)

— Parameter 5: Horizontal Quiet Zone: Q_H

$Q_H \geq 2X$ (see 5.8.3)

— Parameter 6: Vertical Quiet Zone: Q_V

$Q_V \geq 2X$ (see 5.8.3)

— Parameter 7: Width available for the symbol, W

$$W \geq \left(17c + 69\right) + 2Q_H$$

NOTE 1 This parameter could be limited by the scanner field of view or the label width.

— Parameter 8: Height available for the symbol, H

$$H \geq Yr + 2Q_V$$

NOTE 2 This parameter could be limited by the scanner field of view or the label width.

— Parameter 9: Matrix Parameters

$$\left(n + k\right) = \left(c * r\right) < 929$$

— Parameter 10: Symbol Aspect Ratio: A

Before the symbol size can be determined, the number of data codewords and error correction codewords must be calculated. The next step depends upon which parameters are constrained by the application. When the requirements of the application specify an overall symbol aspect ratio, then O.2 gives guidance on calculating the number of data region columns needed to create a symbol of that aspect ratio. If instead the application constrains either the allowed height or width of the symbol (or both), then simpler calculations can be used. O.1 shows this simpler algorithm that can be used when the symbol width is constrained.

— When an overall width W (including quiet zones) is specified, then the number of data columns can be calculated from the equation of Parameter 7 (rounding up to the nearest integer number of columns). The number of rows is then derived from the total number of codewords: $(n+k)=(c*r)$.

— The symbol aspect ratio A is the height to width of the symbol including quiet zones. To achieve a given value of A, the following equation can be solved, with respect to the number of columns (c). The equation assumes that the quiet zones are expressed in precise terms of X but the equation can be used in all cases to produce the best approximation of the number of columns (c).

$$A=\frac{H}{W}=\frac{rY+2Q_V}{17X(c+73)}$$

where

A, c, H, Q_V, r, W, X and Y are as defined in Clause 4;

Q_V is $2X$.

Since the number of rows can be expressed as

$$r=\left(\frac{n+k}{c}\right)$$

where

n and k are as defined in Clause 4.

the equation can be reformulated as:

$$A = \frac{\left(\frac{n+k}{c}\right)Y + 4X}{\left(17c+73\right)x} = \frac{\left(n+k\right)Y + 4cX}{\left(17c^2+73c\right)X} = \frac{\left(n+k\right)\frac{Y}{X} + 4c}{17c^2+73c}$$

Thus:

$$A\left(17c^2+73c\right) - \left[\left(n+k\right)\frac{Y}{X}\right] - 4c = 0$$

This equation can be expressed as:

$$17Ac^2 + \left(73A-4\right)c - \left[\left(n+k\right)Y / X\right] = 0$$

which (substituting x for c) is a quadratic equation in the form of:

$$ax^2 + bx + c = 0$$

Since the solution equation for a quadratic equation is

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

substituting the parameter values of PDF417, the solution equation for the quadratic equation, ignoring the negative value, becomes:

$$c = \frac{-\left(73A-4\right) + \left\{\left(73A-4\right)^2 + 4\left(17A\right)\left[\left(n+k\right)Y / X\right]\right\}^{0,5}}{2\left(17A\right)}$$

The value of n is dependent on the number of pad codewords and this is not known until the matrix parameters are determined. However, the number of source codewords is known. As $m + 1 \leq n$ this can be substituted in the above equation as follows:

$$c = \frac{-\left(73A-4\right) + \left\{\left(73A-4\right)^2 + 4\left(17A\right)\left[\left(m+1+k\right)Y / X\right]\right\}^{0,5}}{2\left(17A\right)}$$

Solving the positive value of c produces a result which is not an integer. The nearest integer value of c gives the best value of the number of columns to achieve the aspect ratio.

The number of rows is given by:

$$r = INT\left[\left(m+1+k\right) / c\right] + 1$$

If$\left(c * r\right) \geq m + 1 + k + c$

then $r = r - 1$

As $\left(c * r\right) = \left(n + k\right)$, the number of pad codewords is $\left(n + k\right) - \left(m + 1 + k\right)$

EXAMPLE To achieve an aspect ratio A = 0,5 for a PDF417 symbol where m + 1 + k = 277, X = 0,33 mm and Y = 1,00 mm

$$c = \frac{-\left[\left(73 * 0,5\right) - 4\right] = \left\{\left[\left(73 * 0,5\right) - 4\right]^2 + 4\left(17 * 0,5\right)\left(\frac{277 * 1,00}{0,33}\right)\right\}^{0,5}}{2\left(17 * 0,5\right)}$$

$$c = \frac{-32,5 + \left(1056 + 28539\right)^{0,5}}{17}$$

$$c = \frac{-32,5 + 172,0}{17}$$

$$c = \frac{139,5}{17} = 8,21 = 8$$

$$r = INT\left(\frac{277}{c}\right) + 1 = INT\left(34,6\right) + 1 = 35$$

$$\left(m + 1 + k\right) \le \left(c * r\right) < 929$$

277 ≤ 280 < 929

The number of pad codewords required is:

$$\left(c * r\right) - \left(m + 1 + k\right)$$

280 – 277 = 3

This symbol has 35 rows and 8 columns and measures 68,97 mm wide by 36,32 mm high, an actual aspect ratio of 0,527.

O.2 Guidelines should any parameters not be achieved

If the symbol fails to conform with the intended label size

a) reduce the data content, if possible,

b) increase the label size in one or both dimensions,

c) reduce the error correction level, and

d) reduce the X-dimension or the module height (Y).

Annex P
(informative)

Calculating the coefficients for generating the error correction codewords – worked example

The following generator polynomial shall be used to calculate coefficients for each error correction level:

$$g_k(x) = (x - 3)\,(x - 3^2)\,(x - 3^3)\ \ldots.\ (x - 3^k)$$

$$= \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \ldots.\ \alpha_{k-1} x^{k-1} + x^k$$

where

$g_k(x)$ is the generator polynomial;

k is the total number of error correction codewords;

α_j is the coefficient of powers of x produced by the generator polynomial $g_k(x)$.

First expand the above equation. Next, calculate the complement of the coefficient from the above.

For $\alpha_j = \alpha_0 \ldots. \alpha_{k-1}$

BEGIN

$\alpha_j = \alpha_j \bmod 929$

END

EXAMPLE

Calculate generator polynomial coefficients for error correction level 1

s	$= 1$	error correction level 1
k	$= 2^{s+1} = 4$	(number of error correction codewords)
$g_4(x)$	$= (x - 3)\,(x - 3^2)\,(x - 3^3)\,(x - 3^4)$	
	$= 59\,049 - 29\,160x + 3\,510x^2 - 120x^3 + x^4$	
α_0	$= 59\,049 \bmod 929$	$= 522$
α_1	$= -29\,160 \bmod 929$	$= 568$
α_2	$= 3\,510 \bmod 929$	$= 723$
α_3	$= -120 \bmod 929$	$= 809$

NOTE Annex F contains all of the coefficient values necessary to encode a PDF417 symbol of any error correction level.

Annex Q
(informative)

Generating the error correction codewords - worked example

To generate the error correction codewords, the algorithm in 5.10 shall be used. (The notation used in the example below is identical to that in 5.10.)

EXAMPLE

The data PDF417 is represented by the codewords 5, 453, 178, 121, 239, when preceded by the Symbol Length Descriptor. There are no pad codewords. Then:

n = 5 (number of codewords including symbol length descriptor)

d_4 = 5

d_3 = 453

d_2 = 178

d_1 = 121

d_0 = 239

Selecting an error correction level of 1 gives:

s = 1

k = 2^{1+1} = 4

$\alpha_0,...,\alpha_3$ = 522, 568, 723, 809

NOTE The example is artificially simple, having only 5 data codewords and 4 error correction codewords. However, it fully illustrates the entire process which expands with increases in the number of data codewords and the number of error correction codewords.

The calculations are:

Initialise $E_0, ..., E_3$ to 0

$t_1 = (d_4 + E_3) \bmod 929 = (5 + 0) \bmod 929 = 5$

$t_2 = (t_1 \times \alpha_3) \bmod 929 = (5 \times 809) \bmod 929 = 329$

$t_3 = 929 - t_2 = 929 - 329 = 600$

$E_3 = (E_2 + t_3) \bmod 929 = (0 + 600) \bmod 929 = 600$

$t_2 = (t_1 \times \alpha_2) \bmod 929 = (5 \text{ x } 723) \bmod 929 = 828$

$t_3 = 929 - t_2 = 929 - 828 = 101$

$E_2 = (E_1 + t_3) \bmod 929 = (0 + 101) \bmod 929 = 101$

$t_2 = (t_1 \times \alpha_1) \bmod 929 = (5 \times 568) \bmod 929 = 53$

$t_3 = 929 - t_2 = 929 - 53 = 876$

$E_1 = (E_0 + t_3) \bmod 929 = (0 + 876) \bmod 929 = 876$

$t_2 = (t_1 \times \alpha_0) \bmod 929 = (5 \times 522) \bmod 929 = 752$

$t_3 = 929 - t_2 = 929 - 752 = 177$

$E_0 = t_3 \bmod 929 = 177 \bmod 929 = 177$

$t_1 = (d_3 + E_3) \bmod 929 = (453 + 600) \bmod 929 = 124$

$t_2 = (t_1 \times \alpha_3) \bmod 929 = (124 \times 809) \bmod 929 = 913$

$t_3 = 929 - t2 = 929 - 913 = 16$

$E_3 = (E_2 + t_3) \bmod 929 = (101 + 16) \bmod 929 = 117$

$t_2 = (t_1 \times \alpha_2) \bmod 929 = (124 \times 723) \bmod 929 = 468$

$t_3 = 929 - t_2 = 929 - 468 = 461$

$E_2 = (E_1 + t_3) \bmod 929 = (876 + 461) \bmod 929 = 408$

$t_2 = (t_1 \times \alpha_1) \bmod 929 = (124 \times 568) \bmod 929 = 757$

$t_3 = 929 - t_2 = 929 - 757 = 172$

$E_1 = (E_0 + t_3) \bmod 929 = (177 + 172) \bmod 929 = 349$

$t_2 = (t_1 \times \alpha_0) \bmod 929 = (124 \times 522) \bmod 929 = 627$

$t_3 = 929 - t_2 = 929 - 627 = 302$

$E_0 = t_3 \bmod 929 = 302 \bmod 929 = 302$

$t_1 = (d_2 + E_3) \bmod 929 = (178 + 117) \bmod 929 = 295$

$t_2 = (t_1 \times \alpha_3) \bmod 929 = (295 \times 809) \bmod 929 = 831$

$t_3 = 929 - t_2 = 929 - 831 = 98$

$E_3 = (E_2 + t_3) \bmod 929 = (408 + 98) \bmod 929 = 506$

$t_2 = (t_1 \times \alpha_2) \bmod 929 = (295 \times 723) \bmod 929 = 544$

$t_3 = 929 - t_2 = 929 - 544 = 385$

$E_2 = (E_1 + t_3) \bmod 929 = (349 + 385) \bmod 929 = 734$

$t_2 = (t_1 \times \alpha_1) \bmod 929 = (295 \times 568) \bmod 929 = 340$

$t_3 = 929 - t_2 = 929 - 340 = 589$

$E_1 = (E_0 + t_3) \bmod 929 = (302 + 589) \bmod 929 = 891$

$t_2 = (t_1 \times \alpha_0) \bmod 929 = (295 \times 522) \bmod 929 = 705$

$t_3 = 929 - t_2 = 929 - 705 = 224$

$E_0 = t_3 \bmod 929 = 224 \bmod 929 = 224$

$t_1 = (d_1 + E_3) \bmod 929 = (121 + 506) \bmod 929 = 627$

$t_2 = (t_1 \times \alpha_3) \bmod 929 = (627 \times 809) \bmod 929 = 9$

$t_3 = 929 - t_2 = 929 - 9 = 920$

$E_3 = (E_2 + t_3) \bmod 929 = (734 + 920) \bmod 929 = 725$

$t_2 = (t_1 \times \alpha_2) \bmod 929 = (627 \times 723) \bmod 929 = 898$

$t_3 = 929 - t_2 = 929 - 898 = 31$

$E_2 = (E_1 + t_3) \bmod 929 = (891 + 31) \bmod 929 = 922$

$t_2 = (t_1 \times \alpha_1) \bmod 929 = (627 \times 568) \bmod 929 = 329$

$t_3 = 929 - t_2 = 929 - 329 = 600$

$E_1 = (E_0 + t_3) \bmod 929 = (224 + 600) \bmod 929 = 824$

$t_2 = (t_1 \times \alpha_0) \bmod 929 = (627 \times 522) \bmod 929 = 286$

$t_3 = 929 - t_2 = 929 - 286 = 643$

$E_0 = t_3 \bmod 929 = 643 \bmod 929 = 643$

$t_1 = (d_0 + E_3) \bmod 929 = (239 + 725) \bmod 929 = 35$

$t_2 = (t_1 \times \alpha_3) \bmod 929 = (35 \times 809) \bmod 929 = 445$

$t_3 = 929 - t_2 = 929 - 445 = 484$

$E_3 = (E_2 + t_3) \bmod 929 = (922 + 484) \bmod 929 = 477$

$t_2 = (t_1 \times \alpha_2) \bmod 929 = (35 \times 723) \bmod 929 = 222$

$t_3 = 929 - t_2 = 929 - 222 = 707$

$E_2 = (E_1 + t_3) \bmod 929 = (824 + 707) \bmod 929 = 602$

$t_2 = (t_1 \times \alpha_1) \bmod 929 = (35 \times 568) \bmod 929 = 371$

$t_3 = 929 - t_2 = 929 - 371 = 558$

$E_1 = (E_0 + t_3) \bmod 929 = (643 + 558) \bmod 929 = 272$

$t_2 = (t_1 \times \alpha_0) \bmod 929 = (35 \times 522) \bmod 929 = 619$

$t_3 = 929 - t_2 = 929 - 619 = 310$

$E_0 = t_3 \bmod 929 = 310 \bmod 929 = 310$

Finally calculate the complement of the results from above, to get the 4 error correction codewords for the encoded data **PDF417** as follows:

$E_3 = 929 - E_3 = 929 - 477 = 452$

$E_2 = 929 - E_2 = 929 - 602 = 327$

$E_1 = 929 - E_1 = 929 - 272 = 657$

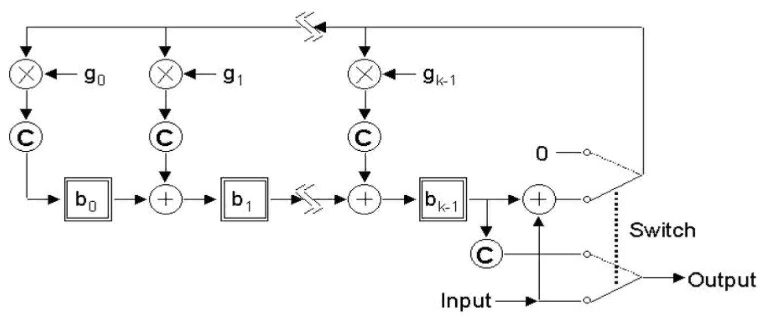
$E_0 = 929 - E_0 = 929 - 310 = 619$

Annex R
(informative)

Division circuit procedure for generating error correction codewords

This procedure is an alternative procedure to that specified in 5.10 and uses a division circuit as the basis of determining the error correction codewords.

The division circuit shall be as illustrated in Figure R.1.



Key

© modulo complement

⊕ modulo addition

⊗ modulo multiplication

Figure R.1 — Error correction codeword encoding circuit

The registers b_0 through to b_{k-1} shall be initialised as zeros. The modulo mathematics shall be defined by the following equations:

$$x \oplus y \equiv (x + y) \bmod 929$$

$$x \otimes y \equiv (x \times y) \bmod 929$$

$$© \, x = (929 - x) \bmod 929$$

where

x and y are numbers from 0 to 928;

$\oplus$ is modulo addition;

$\otimes$ is modulo multiplication;

© is the modulo complement.

There shall be two phases to generate the encoding. In the first phase, with the switch in the down position, the symbol data is passed both to the output and the circuit. The first phase is complete after n clock pulses. In the second phase ($n + 1 \ldots n + k$ clock pulses), with the switch in the up position, the error

correction codewords E_{k-1}, ..., E_0 are generated by flushing the registers in order and complementing the output while keeping the data input at 0.

Annex S
(informative)

Additional guidelines for the use of PDF417

S.1 Autodiscrimination compatibility

PDF417 may be read by suitably programmed bar code decoders which have been designed to autodiscriminate it from other symbologies. The decoder's valid set of symbologies should be limited to those needed by a given application to maximise reading security.

S.2 Pixel-based printing

S.2.1 General principles

Graphics software used to create bar codes on pixel-based printers must scale each bar and space exactly to the pixel pitch of the printer being used. For edge to similar edge decodable symbologies like PDF417, the number of pixels comprising each symbol character must be a fixed and constant integer multiple of the number of modules in the symbol character. For PDF417, the number of modules is 17 for the Start Pattern, other symbol characters, and 18 for the Stop Pattern. Therefore, a given printer can only print a certain set of *X* dimensions.

Compensation for uniform bar width growth (or loss) must be in equal offsetting amounts on all bars and spaces in the symbol. This may be accomplished by changing an integer number of pixels from dark to light or light to dark in the same manner for each bar-space pair in the symbol and for the last bar in the symbol. For example, all pixels along the same edge of every bar in the symbol could be changed from dark to light, or pixels along both edges of every bar in the symbol could be changed from dark to light, provided that the printer resolution is sufficient to allow this to be performed satisfactorily. Any set of dark to light or light to dark pixel changes is acceptable provided the adjustment is performed consistently across the whole symbol and does not change the edge to similar edge measurements or the total symbol character width. Failure to follow these principles results in degraded symbol quality and often results in unreadable symbols.

General purpose printing software designed to support a wide range of printers should provide the user with the capability of adjusting the *X* dimension and bar width growth or loss.

S.2.2 Programmer's Example

These principles can be reduced to the following rules for digital bar code design files.

a) Convert the desired *X* dimension to a module size in pixels rounded down to the nearest integer.

b) Determine the number of pixels corresponding to the desired compensation for uniform bar width growth and round up to the next larger integer.

c) Apply the above results to determine the pixel count of every bar and space in the symbol.

EXAMPLE Using digital bar code design files with a printing device with 24 dots per mm, create a 0,27 mm *X* dimension symbol with 0,06 mm of bar width reduction.

The module size is 24 dots/mm × 0,27 mm/module = 6,5 pixels, which rounds down to 6 pixels per module.

The bar growth compensation is 0,06 mm x 24 dots/mm = 1,4 pixels, which rounds up to 2 pixels.

This process results in the following pixel count for bars and spaces as illustrated in Table S.1.

Table S.1 — Example of correcting pixels for imaging resolution and bar width reduction

Element width (modules)	Nominal width (pixels)	Corrected pixel count	
		Bars	Spaces
1	6	4	8
2	12	10	14
3	18	16	20
4	24	22	26
5	30	28	32
6	36	34	38
8	48	46	n/a

Bibliography

[1] ISO/IEC 8859-1, *Information technology — 8-bit single-byte coded graphic character sets — Part 1: Latin alphabet No. 1*

[2] Blahut, R.E. Theory and Practice of Error Control Codes, published by Addison Wesley, 1984, p. 260, etc.

[3] AIM USA Uniform Symbology Specification PDF417, published by AIM USA, 1994

[4] Europe Uniform Symbology Specification AIM PDF417, published by AIM Europe, 1994

[5] ANSI X3.4, *Coded Character Sets — 7-bit American National Standard Code for Information Interchange (7-bit ASCII) (equivalent to the US national version of ISO/IEC 646)*

[6] AIM Inc. International Technical Standard: ITS/04-001, *Extended Channel Interpretations — Part 1: Identification Schemes and Protocols*1)

[7] AIM Inc. ITS/04-001: International Technical Standard: *Extended Channel Interpretations — Part 2: Registration Procedure for Coded Character Sets and Other Data Formats — Character Set Register*

1) Published by AIM Global, 125 Warrendale-Bayne Road, Suite 100, Warrendale, PA 15086, USA.

ICS 35.040

Price based on 108 pages

CABLED SCANNERS

OPR-2001

Laser Barcode Scanner

HIGHLIGHTS

High performance laser scanner
Compact & sleek design
Hands-free operation
Customizable housing color

EXHIBIT
1006

OPTICON

OPR-2001

Basic product specifications

OPERATING INDICATORS

VISUAL: 1 LED (red/green/orange) NON-VISUAL: Buzzer

OPERATING KEYS

ENTRY OPTIONS: 1 scan key

COMMUNICATION

RS232: DB9 PTF connector with external power supply KEYBOARD WEDGE: MiniDIN6 F/M connector USB: Ver. 1.1, HID/VCP, USB-A connector

POWER

VOLTAGE REQUIREMENT: 5V +- 10% (Keyboard Wedge and USB), 6V (min. 4.5, max. 6.5 V) (RS232) CURRENT CONSUMPTION: Max. 150mA

BARCODE SCANNER OPTICS

LIGHT SOURCE: 650 nm visible laser diode SCAN METHOD: Bi-directional scanning SCAN RATE: 100 scans/sec TRIGGER MODE: Manual, auto-trigger, stand detection READING PITCH ANGLE: -35 to 0˚, 0 to +35˚ READING SKEW ANGLE: -50 to -8˚, +8 to +50˚ READING TILT ANGLE: -20 to 0˚, 0 to +20˚ CURVATURE: R>15 mm (EAN8), R>20 mm (EAN13), MIN. RESOLUTION AT PCS 0.9: 0.127 mm / 5 mil MIN. PCS VALUE: 0.45 DEPTH OF FIELD: At PCS 0.9, Code 39, 40 - 500 mm / 1.57 - 19.69 in (1.0 mm / 39 mil), 20 - 350 mm / 0.79 - 13.78 in (0.5 mm / 20 mil), 20 - 200 mm / 0.79 - 7.87 in (0.25 mm / 10 mil), 20 - 100 mm / 0.79 - 3.94 in (0.15 mm / 6 mil), 30 - 70 mm / 1.18 - 2.76 in (0.127 mm / 5 mil)

SUPPORTED SYMBOLOGIES

BARCODE (1D): JAN/UPC/EAN incl. add on, Codabar/NW-7, Code 11, Code 39, Code 93, Code 128, GS1-128 (EAN-128), GS1 DataBar (RSS), IATA, Industrial 2of5, Interleaved 2of5, ISBN-ISMN-ISSN, Matrix 2of5, MSI/Plessey, S-Code, Telepen, Tri-Optic, UK/Plessey POSTAL CODE: Chinese Post, Korean Postal Authority code 2D CODE: Composite codes, MicroPDF417, PDF417

DURABILITY

TEMPERATURE IN OPERATION: -5 to 50˚C / 23 to 122˚F TEMPERATURE IN STORAGE: -20 to 60˚C / -4 to 140˚F HUMIDITY IN OPERATION: 20 - 85% (non-condensing) HUMIDITY IN STORAGE: 10 - 90% (non-condensing) AMBIENT LIGHT IMMUNITY: Fluorescent 3,000 lx max, Direct sun 50,000 lx max, Incandescent 3,000 lx max DROP TEST: 1.5 m / 5 ft drop onto concrete surface VIBRATION TEST: 10 - 100 Hz with 2G for 1 hour PROTECTION RATE: IP 42

PHYSICAL

DIMENSIONS (W X H X D): 56 x 151 x 31 mm / 2.20 x 5.94 x 1.22 in DIMENSIONS STAND (W X H X D): 100 x 175 x 127 mm / 3.94 x 6.89 x 5.00 in (excl. scanner) WEIGHT BODY: Ca. 60 g / 2.1 oz (excl. cable) WEIGHT STAND: Ca. 115 g / 4.1 oz CASE: ABS, any color available, chrome stand

REGULATORY & SAFETY

PRODUCT COMPLIANCE: CE, FCC, VCCI, RoHS, JIS-C-6802 Class 2, IEC 60825-1 Class 2, FDA CDRH Class II

ENCLOSED ITEMS

Stand (optionally sold separately)

SOLD SEPARATELY

POWER SUPPLY: 100-240V/0.5A, 50/60 Hz, 6V/2A (for RS232)

MODELS

INTERFACE VERSIONS: RS232, Keyboard Wedge, USB

_OPR-2001_001

WWW.OPTICON.COM

US010140490B2

(12) **United States Patent**
Kubo et al.

(10) **Patent No.: US 10,140,490 B2**
(45) **Date of Patent: Nov. 27, 2018**

(54) **MODULE FOR OPTICAL INFORMATION READER**

(71) Applicant: **OPTOELECTRONICS CO., LTD.**, Warabi-shi, Saitama (JP)

(72) Inventors: **Wataru Kubo**, Warabi (JP); **Satoshi Komi**, Warabi (JP)

(73) Assignee: **OPTOELECTRONICS CO., LTD.**, Warabi-shi (JP)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **15/049,691**

(22) Filed: **Feb. 22, 2016**

(65) **Prior Publication Data**

US 2016/0253536 A1 Sep. 1, 2016

Related U.S. Application Data

(60) Provisional application No. 62/121,312, filed on Feb. 26, 2015.

(30) **Foreign Application Priority Data**

Oct. 6, 2015 (JP) 2015-198590

(51) **Int. Cl.**
G06K 7/10 (2006.01)
G02B 26/10 (2006.01)
(Continued)

(52) **U.S. Cl.**
CPC ***G06K 7/10831*** (2013.01); ***G02B 7/025*** (2013.01); ***G02B 7/04*** (2013.01);
(Continued)

(58) **Field of Classification Search**
CPC G02B 26/105; G02B 7/023; G02B 7/025; G02B 27/30; G02B 27/62; G06K 7/1098; G06K 7/10831
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,918,702 A * 4/1990 Kimura G02B 6/4204 372/101
5,490,158 A * 2/1996 Mogi B41J 2/471 372/36
(Continued)

FOREIGN PATENT DOCUMENTS

JP 05022530 A * 1/1993
JP 2000028888 A * 1/2000
(Continued)

OTHER PUBLICATIONS

English Translation of JP2010117510; retrieved Jan. 23, 2017.*
(Continued)

Primary Examiner — Christopher Stanford
(74) *Attorney, Agent, or Firm* — Westerman, Hattori, Daniels & Adrian, LLP

(57) **ABSTRACT**

A collimator lens unit in which an aperture limit stop formation member and a collimator lens are integrally disposed in a cylindrical member is inserted in a lens-barrel hole of the module casing so as to be reciprocatable in an optical axis direction, and a light-emitting unit is fixed in the lens-barrel hole, with an optical axis of a light source aligned with an optical axis of the collimator lens. A long hole through which an adjust pin is penetrated so as to be reciprocatable in the optical axis direction is formed in a peripheral sidewall of the lens-barrel hole, and a fitting portion in which the adjust pin is fit is formed in an outer peripheral surface of the cylindrical member. On an inner peripheral surface of the lens-barrel hole, at a position opposed to the fitting portion, bearing portions in contact with the outer peripheral surface of the cylindrical member are formed.

8 Claims, 6 Drawing Sheets

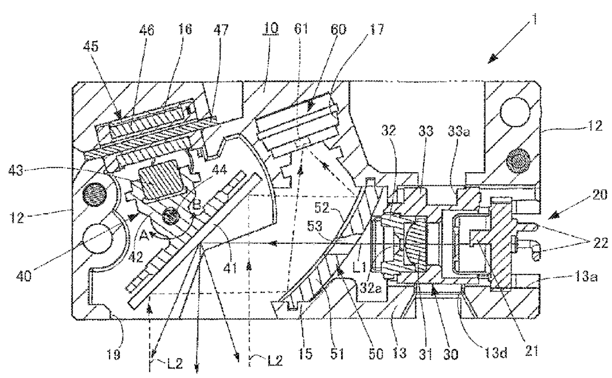


EXHIBIT
1007

(51) **Int. Cl.**
G02B 27/30 (2006.01)
G02B 7/04 (2006.01)
G02B 7/02 (2006.01)

(52) **U.S. Cl.**
CPC ***G02B 26/105*** (2013.01); ***G02B 27/30*** (2013.01); ***G06K 7/1098*** (2013.01)

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,178,016	B1 *	1/2001	Ashe	H04N 1/03 358/487
6,410,904	B1	6/2002	Ito et al.	
6,661,588	B1 *	12/2003	Huang	G02B 7/026 359/819
7,206,109	B2	4/2007	Hayakawa et al.	
2004/0109243	A1 *	6/2004	Orimo	G02B 7/023 359/819
2005/0040324	A1 *	2/2005	Kobayashi	G02B 7/025 250/234
2006/0204234	A1 *	9/2006	Kawai	G02B 7/10 396/72
2007/0047110	A1 *	3/2007	Matsushima	G02B 7/025 359/819
2011/0013901	A1 *	1/2011	Utsugi	G02B 7/026 396/529

FOREIGN PATENT DOCUMENTS

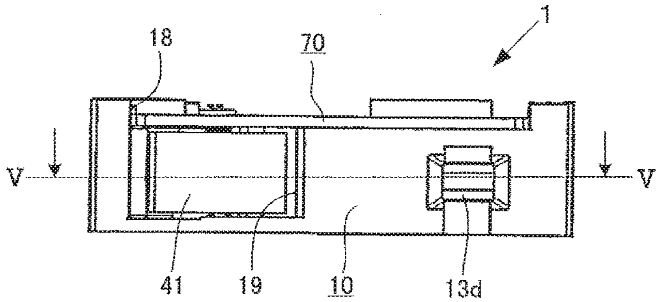
JP	2001-215425	A		8/2001
JP	2003-76942	A		3/2003
JP	2006138914	A	*	6/2006
JP	2010-117510	A		5/2010
WO	03/019463	A1		3/2003

OTHER PUBLICATIONS

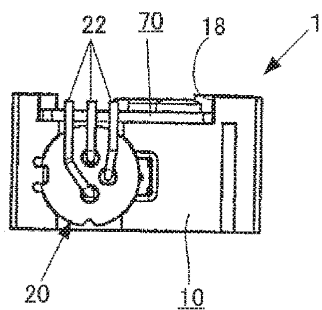
Certificate of Patent dated May 27, 2016, issued in counterpart Japanese Patent Application No. 2015-198590. (1 page).
English translation of Notification of Reasons for Refusal dated Feb. 23, 2016, issued in counterpart Japanese Patent Application No. 2015-198590 (3 pages).
Office Action dated Feb. 23, 2016, issued in counterpart Japanese Patent Application No. 2015-198590. (3 pages).

* cited by examiner

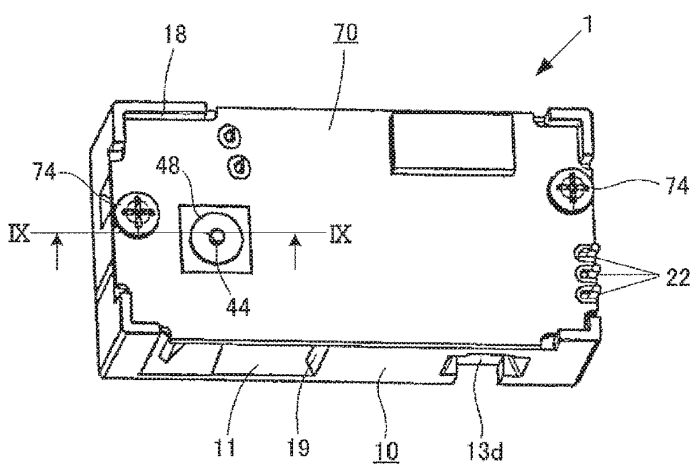
[Fig. 1]



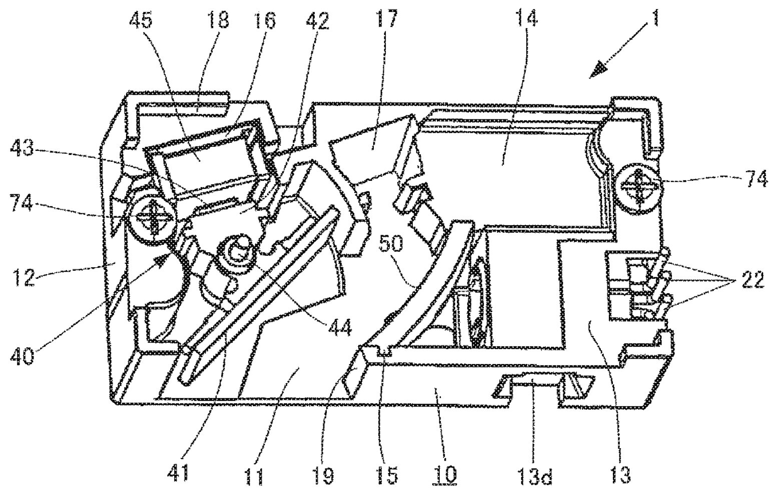
[Fig. 2]



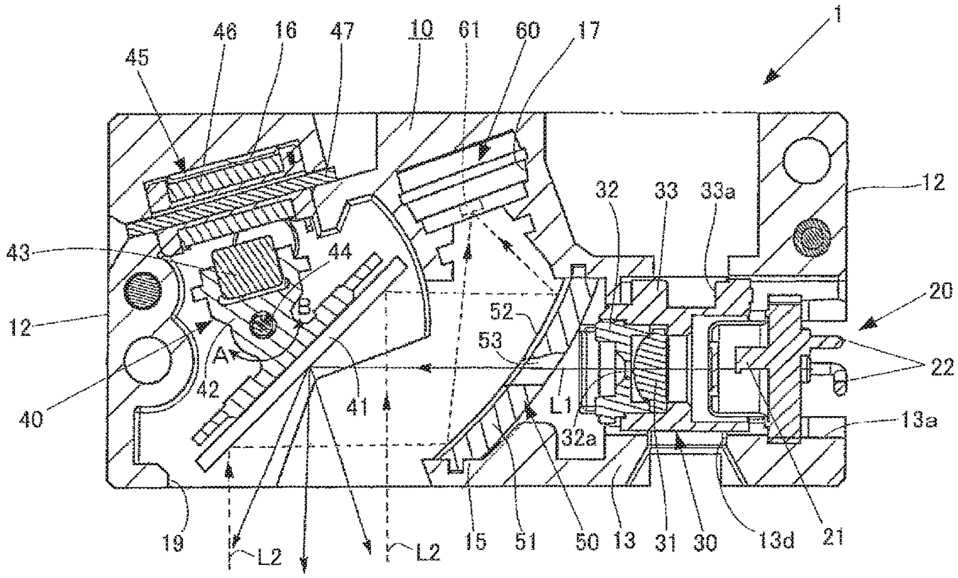
[Fig. 3]



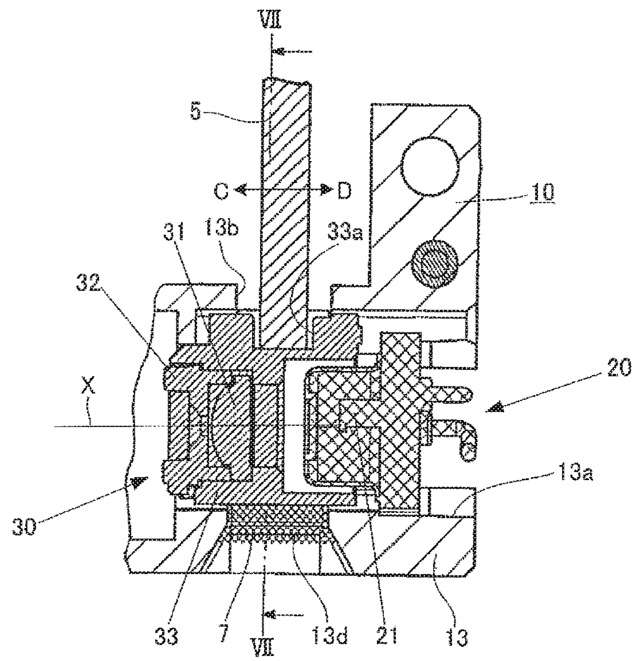
[Fig. 4]


45
18
16
42
17
14
1
43
74
12
40
50
44
74
22
41
11
19
15
10
13d
13

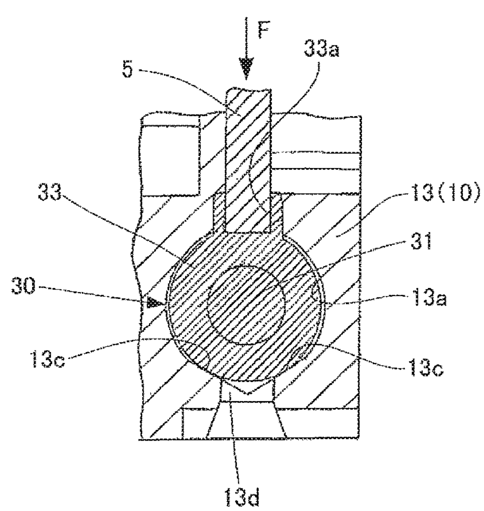
[Fig. 5]


45
46
16
47
10
61
60
17
1
43
12
40
44
B
A
42
41
32
33
33a
12
20
52
53
L1
32a
22
13a
19
L2
L2
15
51
50
13
31
30
13d
21

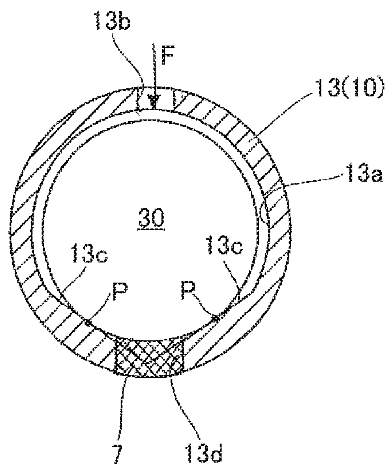
[Fig. 6]


VII
5
C
D
10
31
13b
33a
32
20
X
13a
30
33
7
VII
13d
21
13

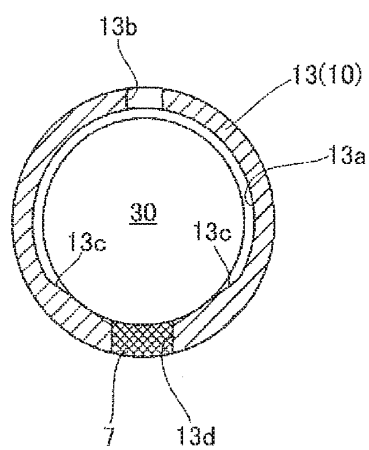
[Fig. 7]


F
33a
5
33
13(10)
31
30
13a
13c
13c
13d

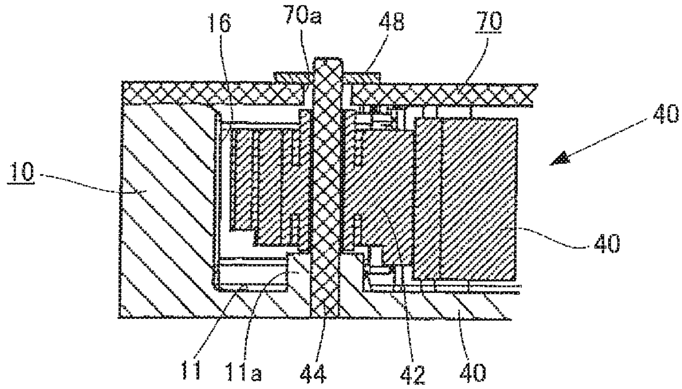
[Fig. 8A]



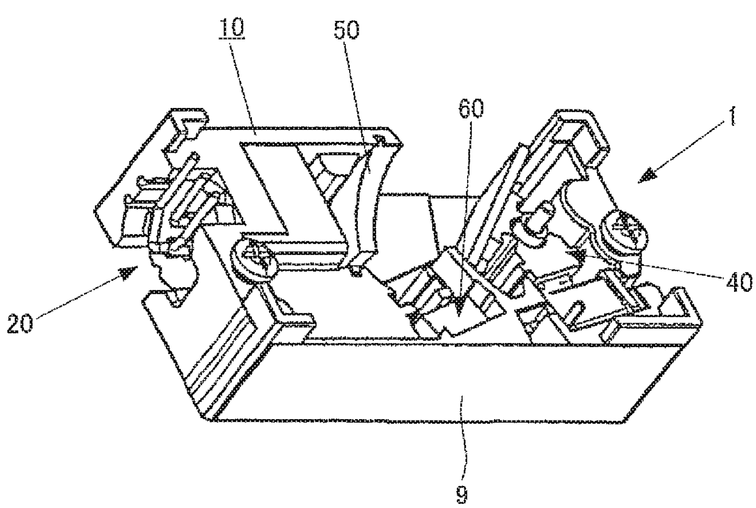
[Fig. 8B]



[Fig. 9]

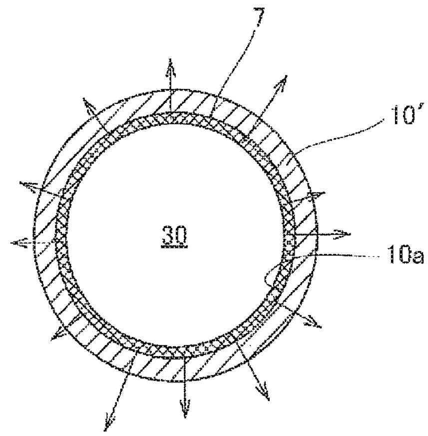


[Fig. 10]



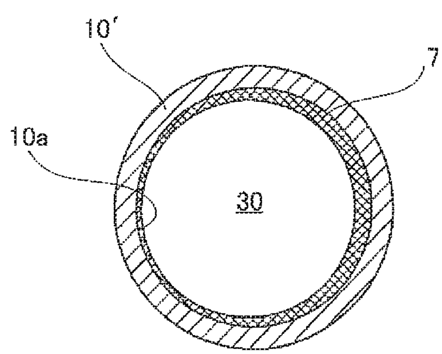
[Fig. 11A]

PRIOR ART



[Fig. 11B]

PRIOR ART



MODULE FOR OPTICAL INFORMATION READER

FIELD OF THE INVENTION

The invention relates to a module installed in an optical information reader for reading optical information of a bar code and the like.

BACKGROUND OF THE INVENTION

As an optical information reader, bar code readers which read optical information of bar codes, two-dimensional codes, and the like indicating information such as names and prices of products are used widely by the distribution industry and the retail industry.

The bar code readers are roughly classified into hand-held ones held by one hand when in use and stationary ones, and the hand-held ones further include a pen type, a touch type, and a light beam scanning type (laser type). Among these, an optical information reader being an object of the invention. is an optical information reader such as a hand-held bar code reader of the light beam scanning type.

A bar code reader of the light beam scanning type shapes light emitted by a light source such as a laser diode (semiconductor laser) into a beam, deflects the light beam by a mirror so that the light beam hits on a bar code, and while rotating or vibrating (swinging) the mirror, scans the bar code so that the light beam moves across the bar code.

Then, the reflected light from the bar code is condensed, is received by a light-receiving sensor, and is converted to an electrical signal. The electrical signal is coded after A/D conversion and the resultant is output as bar code read information. In the hand-held optical information reader of the light beam scanning type, its read engine part is required to be greatly reduced in size and weight.

Under such circumstances, there has come into use a module for an optical information reader in which the aforesaid light source, a collimator lens for shaping the light emitted by the light source into a beam, a vibration mirror and it's driver, a collector mirror or a condenser lens, a light-receiving sensor, a processing circuit for a detection signal of the light-receiving sensor, and so on are assembled in a common easing to be modularized, as described in, for example, PLT 1, 2, 3, and so on.

In such a module for an optical information reader, a light-emitting unit whose light source is, for example, a laser diode, the collimator lens for turning the light emitted by the light source into a parallel luminous flux, and a member having an aperture through which the parallel luminous flux exits as a thin beam need to be fixedly positioned in a lens barrel, with their optical axes aligned. Further, in order for the collimator lens to surely generate the light beam which is to be converged in the parallel light flux or at a finite distance, it is necessary to accurately adjust the distance between the light-emitting unit and the collimator lens (collimation adjustment or focus adjustment) so that a focal point of the collimator lens and a light-emitting point of the light-emitting unit have a predetermined positional relation.

Therefore, in a light beam generating part in the module for the optical information reader disclosed in the aforesaid PLT 1, 2, 3, in part of the module casing, a lens-barrel hole is provided, at whose leading end portion the aperture being an aperture limit stop for letting the light beam exit therethrough is formed and whose rear end portion is opened to be formed as a press-fitting portion where to press-fit the light-emitting unit. Then, the collimator lens is bonded and fixed at a position short of the aperture, at a leading end rear side portion of the lens-barrel hole, and the light-emitting unit is pressed into the press-fitting portion from the rear end portion, whereby they are positioned.

CITATION LIST

Patent Literature

{PTL 1} U.S. Pat. No. 7,206,109 B2
{PTL 2} WO 03/019463 A1
{PTL 3} JP 2003-76942 A

SUMMARY OF INVENTION

Technical Problem

Such a conventional module for an optical information reader had the following problems.

Since the light-emitting unit is pushed into the lens-barrel hole while being pressed, and for the collimation adjustment, it is moved in one direction (a direction which is an optical axis direction and in which it approaches the collimator lens), re-adjustment by returning it was not possible. Accordingly, when the light-emitting unit is pushed too much, this module becomes a defective product, resulting in worsened production yields.

Further, delicate adjustment on a micron level was not possible since the light-emitting unit is press-fit and thus frictional resistance is high, and there is a possibility that the collimator lens and the laser diode are tilted relatively to each other.

Object of the Invention

The invention was made in consideration of the above technical background, and has an object to make it possible to, in a module for an optical information reader, easily and delicately make collimation adjustment and focus adjustment in both directions of back and forth directions along an optical axis while preventing a collimator lens and a laser diode from being tilted relatively to each other, thereby greatly reducing the occurrence of defective products to enhance production yields.

Solution to Problem

A module for an optical information reader according to the invention is a module for an optical information reader in which a light-emitting unit having a light source such as a laser diode, a collimator lens, a vibration mirror for scanning, a collector mirror or a condenser lens, and a light-receiving sensor are disposed in a module casing to be modularized, and it is structured as follows in order to achieve the aforesaid object.

A collimator lens unit in which an aperture limit stop formation member and the collimator lens are integrally disposed in a cylindrical member is inserted in a lens-barrel hole of the module casing so as to be reciprocatable in an optical axis direction within a predetermined range, and the light-emitting unit is fixed to the module easing in the lens-barrel hole, with an optical axis of the light source aligned with an optical axis of the collimator lens.

Further, a long hole through which an adjust pin is penetrated so as to be reciprocatable in the optical axis direction within a predetermined range is formed in a peripheral sidewall of the lens-barrel hole of the module

casing, and a fitting portion in which a tip portion of the adjust pin penetrating through the long hole is fit is formed in an outer peripheral surface of the cylindrical member.

Furthermore, on an inner peripheral surface of the lens-barrel hole of the module casing, at or near a position opposed to the fitting portion, bearing portions in contact with the outer peripheral surface of the cylindrical member are formed at positions symmetrical with respect to the position in terms of an inner circumferential direction of the lens-barrel hole.

Preferably, the bearing portions form a V-shaped slope by two flat surfaces which, in a circumferential direction, are in point contact with the outer peripheral surface of the cylindrical member, and in an axial direction, are in line contact with the outer peripheral surface.

Preferably, an open-hole through which an adhesive for fixing the cylindrical member is tillable is formed in a middle region, of the V-shaped slope, which is not in contact with the outer peripheral surface of the cylindrical member.

The collimator lens unit may be structured such that the collimator lens and the aperture limit stop formation member are fixed to the cylindrical member.

The collimator lens unit may be structured such that the aperture limit stop formation member and the cylindrical member are integrally disposed on the collimator lens itself.

The collimator lens unit may be structured such that the collimator lens and the cylindrical member are integrally formed of the same material or different kinds of materials, and to the resultant formed body, the aperture limit stop formation member is fixed.

The module casing may be formed of resin. Desirably, the resin is reinforced resin in which carbon is dispersed.

Desirably, a metallic foil is affixed on an outer wall surface of the module casing made of the resin, at least near a portion where the light-receiving sensor is housed.

Advantageous Effects of Invention

In the module for the optical information reader according to the invention, the module casing is fixed to a jig, the tip portion of the adjust pin is passed through the long hole of the module casing to be inserted into the lens-barrel hole and is fit in the fitting portion of the cylindrical member or the cylindrical part of the collimator lens unit, and when the adjust pin is moved in the optical axis direction of the collimator lens by a linear movement mechanism of the jig, it is possible to move the collimator lens integrally with the cylindrical member or the cylindrical part to easily make collimation adjustment.

Further, the adjustment can be made while the collimator lens is moved in the both directions of the optical axis direction, that is, directions in which it approaches and separates from the light source, and therefore, if it is moved too much in one of the directions, the re-adjustment is possible by returning it. This can greatly reduce the occurrence of defective products to enhance production yields. Setting a movement pitch of the adjust pin fine can facilitate, even delicate and highly accurate adjustment.

Since it is possible for the adjust pin to move the cylindrical member or the cylindrical part of the collimator lens unit while pressing it against the bearing portions on the opposite side, the adjustment can be made without any deviation of the optical axis. Further, by reducing a contact area between the outer peripheral surface of the cylindrical member or the cylindrical part and the bearing portions, it is possible to move the cylindrical member or the cylindrical part smoothly with a relatively small force.

If the open-hole from which the adhesive can be filled is formed near the bearing portions, it is possible to fill the adhesive from the opening to fixedly bond the cylindrical member or the cylindrical part to the module casing after the collimation adjustment while pressing the cylindrical member or the cylindrical part against the bearing portions by the adjust pin, and thus adjustment deviation and optical axis deviation of the collimator lens are not liable to occur.

Forming the module casing of the resin having the heat dissipation property and the shielding property makes it possible to reduce weight as well as to reduce cost far more than forming it of metal. In addition, the heat dissipation property and the shielding property to a degree not practically problematic can also be obtained. Using the black resin such as the reinforced resin in which carbon is dispersed makes it possible to prevent the reflection of the light. If the shielding effect is not sufficient, by affixing the metallic foil on a necessary portion of the outer wall surface of the module casing, it is possible to enhance the shielding effect.

BRIEF DESCRIPTION OF DRAWINGS

FIG. **1** is a front view of one embodiment of a module for an optical information reader according to the invention.

FIG. **2** is a right side view of the module for the optical information reader according to the same.

FIG. **3** is a perspective view of the module for the optical information reader according to the same seen from obliquely above.

FIG. **4** is a perspective view of the module for the optical information reader according to the same seen from the same direction as that in FIG. **3**, with a circuit board removed.

FIG. **5** is an enlarged sectional view taken along V-V line in FIG. **1**.

FIG. **6** is an enlarged partial sectional view illustrating a light beam generating part in FIG. **5** together with an adjust pin.

FIG. **7** is a partial sectional view taken along VII-VII line in FIG. **6**.

FIG. **8**A is a sectional view schematically illustrating a cross section similar to that in FIG. **7** to explain characteristics of collimation adjustment according to an embodiment of the invention.

FIG. **8**B is a sectional view illustrating the same in a state after an adhesive for fixing a collimator lens unit to a module casing is cured.

FIG. **9** is a sectional view taken along IX-IX line in FIG. **3**.

FIG. **10** is a perspective view illustrating an embodiment in which a metallic foil is affixed on an outer wall surface of the module casing to enhance a shielding effect, with the circuit board removed.

FIG. **11**A is a sectional view illustrating a bonding example being a reference example for comparison with the embodiment of the invention, which corresponds to FIG. **8**A before the adhesive is cured.

FIG. **11**B is a sectional view illustrating the same, which corresponds to FIG. **8**B after the adhesive is cured.

DETAILED DESCRIPTION

Hereinafter, modes for carrying out the invention will be described based on the drawings.

First, the entire structure of one embodiment of a module for an optical information reader according to the invention will be specifically described with reference to FIG. **1** to FIG. **5**.

FIG. **1** is a front view of the module for the optical information reader and FIG. **2** is a right side view thereof FIG. **3** is a perspective view of the module for the optical information reader seen from obliquely above, and FIG. **4** is a perspective view of the same seen from the same direction, with a circuit board removed. FIG. **5** is an enlarged sectional view taken along V-V line in FIG. **1**.

This module **1** for the optical information reader is a read engine installed in an optical information reader such as a bar code reader, and as illustrated in these drawings, it is composed of a module casing **10**; a light-emitting unit **20**, a collimator lens unit **30**, a vibration mirror driver **40**, a collector mirror **50** having a concave surface shape, and a light-receiving sensor **60** which are assembled in the module casing **10**; a circuit board **70** attached to an upper surface of the module casing **10**; and so on.

The module casing **10** has, for example, a size of 14 mm depth (D) 28 mm width (W), and 7.5 mm height (H) as its whole outer shape, but this is not restrictive. Since such a module casing is required to have a heat dissipation property and a shielding property, it has been conventionally formed by a die casting manufacturing method by using metal such as, for example, zinc called ZDC2 or a magnesium alloy called AZ91D. The module casing of the module for the optical information reader according to the invention may similarly be formed of metal by the die casting manufacturing method.

However, the module casing **10** in this embodiment is formed of resin higher in heat dissipation property (thermal conductivity) and shielding property (electric conductivity) than ordinary resin, for example, formed of black reinforced resin in which carbon is dispersed. As a specific example of the resin material, TCF1140 manufactured by Mitsubishi Engineering-Plastics Corporation is preferably used. Forming the module casing **10** of such resin can achieve a cost reduction and a great weight reduction, and to obtain a heat dissipation property and a shielding property high enough for practical application.

Further, this module casing **10** has a bottom surface portion **11**, a sidewall portion **12** surrounding its periphery, a light beam generating part housing part **13**, a LSI housing recessed part **14**, a collector mirror attachment part **15**, a vibration mirror driver attachment part **16**, a light-receiving unit attachment part **17**, a circuit board holding part **18**, and so on.

On the bottom surface portion **11** of the vibration mirror driver attachment part **16**, a boss **11***a* (refer to FIG. **9**) is formed, and a lower end portion of a support shaft **44** of a vibration mirror **41** is fit therein to be supported. A front face, of the sidewall portion **12**, corresponding to the vibration mirror driver attachment part **16** is opened to form an opening **19** for letting a light beam exit and incident.

As illustrated in FIG. **5**, the light-emitting unit **20** and the collimator lens unit **30** which form a light beam generating part are disposed in a lens-barrel hole **13***a* which is formed in the light beam generating part housing part **13** of the module casing **10** and which has a cylindrical inner peripheral surface.

The light-emitting unit **20** has a laser diode **21** as a light source, and is inserted to be fixed in the lens-barrel hole **13***a* from an opening of the sidewall portion **12** of the module casing **10** on the right side in FIG. **5**. As illustrated in FIG. **2** and so on, three terminals **22** of the laser diode **21** project to extend upward from a rear end surface of the light-emitting unit **20** and are connected to terminals of the board **70** side.

In the collimator lens unit **30**, a collimator lens **31** and an aperture limit stop formation member **32** in which an aperture **32***a* being an aperture limit stop is formed are integrally fixed in a cylindrical member **33** being a collimator lens barrel. The aperture limit stop formation member **32** is fixed to a front end portion of the cylindrical member **33** by an adhesive or the like, presses the collimator lens **31** against an inner periphery stepped portion of the cylindrical member **33** to fix it, and has the aperture **32***a* disposed just in front of the collimator lens **31**.

This collimator lens unit **30** is inserted in the lens-barrel hole **13***a* of the light beam generating part housing part **13** of the module casing **10** so as to be reciprocatable in an optical axis direction within a predetermined range. The aperture limit stop formation member **32** and the cylindrical member **33** can be formed of the same material as or has performance equivalent to that of the module casing **10** (polycarbonate containing 20% glass, aluminum, or the like). The light-emitting unit **20** is fixed to the module casing **10** in the lens-barrel hole **13***a* so as to partly enter the inside of the cylindrical member **33**, with an optical axis of the light source being aligned with an optical axis of the collimator lens **31**. Details of the collimator lens unit **30** and collimation adjustment will be described later.

As illustrated in FIG. **4** and FIG. **5**, the vibration mirror driver **40** is composed of: a vibration mirror **41** for light beam scanning made of metal, resin, or glass; a vibration mirror holding member **42** fixed to a front surface portion of the vibration mirror **41** and made of resin; a movable magnet (permanent magnet) **43** fixed to a rear surface side of the vibration mirror holding member **42**; the support shaft **44** in a pin shape supporting the vibration mirror holding member **42** so as to allow its rotation; and a coil unit **45** disposed to face and to be apart from and in parallel to the movable magnet **43**. In the coil unit **45**, a yoke **47** penetrates through a coil **46** in a direction perpendicular to a winding direction of the coil **46**.

These are attached to the vibration mirror driver attachment part **16** of the module casing **10**. Then, by an action of the movable magnet **43** and the coil unit **45**, the vibration mirror holding member **42** and the vibration mirror **41** fixed thereto are vibrated (swung) in a seesaw manner as indicated by the arrows A, B in FIG. **5**.

A collector mirror **50** having a concave surface shape is fixed in a tilting manner to the collector mirror attachment part **15** of the module casing **10** so as to face the vibration mirror **41** and the light-receiving sensor **60**. The collector mirror **50** has a reflective film **52** formed on a concave curved surface of a curved substrate **51** made of resin, and has a rectangular or circular through hole **53** formed at its center portion to allow the light beam to pass therethrough.

The light-receiving sensor **60** has a light-receiving element **61** such as a photodiode (PD), and is integrated with the circuit board **70** with its two terminals connected to the circuit board **70** illustrated in FIG. **3**. Accordingly, when the circuit board **70** is mounted on the circuit board holding part **18** of the module casing **10**, the light-receiving sensor **60** is inserted to the light-receiving sensor attachment part **17** to be disposed at a predetermined position.

On the circuit board **70**, a not-illustrated necessary wiring pattern is formed and various kinds of chip-shaped electronic components are attached, and on its rear surface side, a LSI (large-scale integrated circuit) playing a central role in signal processing and control is mounted.

Then, this circuit board **70** is fixedly attached to the upper surface of the module casing **10** with a plurality of screws **74**, and serves also as an upper cover of this module **1** for

the optical information reader. At this time, the LSI mounted on the rear surface is housed in the LSI housing recessed part **14** (FIG. **4**) of the module casing **10**. The LSI is prevented from being influenced by electromagnetic wave noise generated by other electronic devices, cellular phones, and so on since four surfaces of its outer periphery are surrounded by the resin with a high shielding property of the module casing **10**.

Functions of the module **1** for the optical information reader thus structured will be described by mainly using FIG. **5**.

A laser ray is generated as a result of the light emission of the laser diode **21** being the light source in the light-emitting unit **22**, this is turned into a luminous flux which is parallel or is converged at a desired distance by the collimator lens **31**, and the luminous flux is passed through the aperture **32***a* to be radiated as a laser beam L**1** indicated by the solid line.

This laser beam L**1** passes through the through hole **53** of the collector mirror **50** to reach the vibration mirror **41**, is reflected in a predetermined angular range whose center is 90°, due to the vibration of the vibration mirror **41**, and exits from the opening **19** to the outside. This laser beam irradiates a not-illustrated bar code symbol.

The bar code symbol has a plurality of black and white vertical stripes each having a predetermined width stipulated by the standard as is well known. They are called black bars and spaces. Light with different reflectance is reflected depending on a lateral width of each of the black bars and the spaces.

Rays L**2** (indicated by the broken-line arrows in FIG. **5**) reflected from the bar code symbol pass through the opening **19** again and enter the vibration mirror **41** to be reflected. Their reflected lights are collected by the collector mirror **50**. At this time, since the vibration mirror **41** vibrates due to a magnetic force generated between the coil unit **45** and the movable magnet **43**, it is possible for the lights in a wide range reflected from the bar code symbol to enter and to be sent to the collector mirror **50**. Then, the lights collected by the collector mirror **50** are all received by the light-emitting element **61** of the light-receiving sensor **60**.

The light-receiving sensor **60** outputs an electrical signal according to the intensity of the light received by the light-receiving element **61** and sends the electrical signal to the circuit board **70**. In the circuit board **70**, the electrical signal is A/D converted and thereafter the digital signal is processed, whereby data read from the bar code symbol is obtained.

By assembling this module **1** for the optical information reader in a not-illustrated case together with a power supply part and so on, it is possible to easily complete a compact optical information reader such as a hand-held bar code reader.

Next, a characterizing structure in the light beam generating part of this module **1** for the optical information reader, a method of the collimation adjustment (also called focus adjustment), and a fixing method of the collimator lens unit **30** and the module casing **10** after this adjustment will be described based on FIG. **6**, FIG. **7**, FIG. **8**A, and FIG. **8**B.

FIG. **6** is an enlarged partial sectional view illustrating the light beam generating part in FIG. **5** together with the adjust pin, and FIG. **7** is a partial sectional view taken along VII-VII line in FIG. **6**. In these drawings, the light-emitting unit **20** is entirely cross-hatched in the same manner, and the collimator lens unit **30** is also entirely hatched in the same manner.

FIG. **8**A is a sectional view schematically illustrating a cross section similar to that in FIG. **7** to explain characteristics of the collimation adjustment according to the embodiment of the invention, and FIG. **8**B is a sectional view illustrating a state after the adhesive for fixing this collimator lens unit **30** to the module casing is cured. In these drawings, the hatching of the collimator lens unit **30** is omitted.

As previously described, the collimator lens unit **30** has the collimator lens **31** and the aperture limit stop formation member **32** integrally fixed in the cylindrical member **33**, and is inserted in the lens-barrel hole **13***a* of the light beam generating part housing part **13** of the module casing **10** so as to be reciprocatable in the direction along the optical axis X within the predetermined range. Thus, an outside diameter of the cylindrical member **33** is slightly smaller than an inside diameter of the lens-barrel hole **13***a* and there exists a small gap between the both.

In a peripheral sidewall of the lens-barrel hole **13***a* in the light beam generating part housing part **113** of the module casing **10**, a long hole **131**) through which the adjust pin **5** penetrates so as to be reciprocatable in the optical axis direction (C and D directions indicated by the arrows in FIG. **6**) within a predetermined range is formed. Further, in an outer periphery of the cylindrical member **33** of the collimator lens unit **30**, a fitting portion **33***a* in which a tip portion of the adjust pin **5** penetrating through the long hole **13***b* is fit is formed. The fitting portion **33***a* is a recessed portion in this embodiment, but if a recessed portion is formed in a tip surface of the adjust pin **5**, the fitting portion of the outer periphery of the cylindrical member **33** can be a projecting portion.

On an inner peripheral surface of the lens-barrel hole **13***a* of the module casing **10**, at or near a position opposed to the fitting portion **33***a*, bearing portions **13***c* in contact with the outer peripheral surface of the cylindrical member **33** are disposed at positions symmetrical with respect to this position in terms of a circumferential direction of the lens-barrel hole **13***a*.

In this embodiment, as illustrated in FIG. **7**, the pair of bearing portions **13***c* form a V-shaped slope which is thick so as to make an inside diameter of the inner peripheral surface of the lens-barrel hole **13***a* smaller than that at the other portion of the inner peripheral surface. Accordingly, when the collimator lens unit **30** is pressed in the arrow F direction in FIG. **7** by the adjust pin **5**, the bearing portions **13***c* come into point contact with the outer peripheral surface of the cylindrical member **33** at P points illustrated in FIG. **8**A in the circumferential direction and into line contact therewith in an axial direction. In this state, the optical axis X of the collimator lens and a light emission center of the light-emitting unit **20** coincide with each other, and the collimator lens is easily movable along the optical axis X as it is.

These bearing portions **13***c* are not limited to the V-shaped slope, but the bearing portion may be a slightly inwardly projecting curved surface formed at part of the inner peripheral surface of the lens-barrel hole **13***a,* or a curved-surface projection provided along the axial direction on the inner peripheral surface.

Further, in the bearing portions **13***c* or in the vicinity thereof, in this embodiment, in a middle region, of the V-shaped slope being the pair of bearing portions **13***c,* which is not in contact with the outer peripheral surface of the cylindrical member **33**, an open-hole **13***d* through which an adhesive for fixing the cylindrical member **33** is finable is formed.

The collimation adjustment (focus adjustment) in the light beam generating part of the module **1** for the optical information reader thus structured is performed as follows.

The module **1** for the optical information reader whose assembly is finished is fixed to a not-illustrated jig. Then, as illustrated in FIG. **6**, the tip portion of the adjust pin **5** disposed on the jig is inserted to the long hole **13***b* formed in the light beam generating part housing part **13** of the module casing **10**, and further is fit in the fitting portion **33***a* formed in the outer periphery of the cylindrical member **33** of the collimator lens unit **30**. By the adjust pin **5** being pressed in the arrow F direction illustrated in FIG. **7**, the cylindrical member **33** of the collimator lens unit **30** is pressed in the same direction, and its outer peripheral surface is brought into point contact with the pair of bearing portions **13***c* formed on the inner peripheral surface of the lens-barrel hole **13***a,* at the P points illustrated in FIG. **8**A.

When the adjust pin **5** is moved in this state in the arrow C direction or D direction illustrated M FIG. **6** by a linear feeding mechanism of the jig, the collimator lens unit **30** moves in accordance therewith along the optical axis X of the collimator lens **31**, so that its distance from the light-emitting unit **20** fixed in the lens-barrel hole **13***a* of the module casing **10** changes.

Consequently, the laser light emitted from the laser diode **21** being the light source of the light-emitting unit **20** is turned into an accurate parallel luminous flux by the collimator lens **31** of the collimator lens unit **30**, which makes it possible to perform the collimation adjustment or the focus adjustment so that the light exits as a prescribed laser beam through the aperture **32***a.*

This adjustment can be performed by moving the collimator lens unit **30** in both a direction in which it separates from the light-emitting unit **20** and a direction in which it approaches the light-emitting unit **20**, and therefore in a case where the adjustment is made excessively in one of the directions, it is possible to return it for re-adjustment, so that a defective product due to the poor collimation adjustment scarcely occurs. The adjust pin **5** can be moved by a linear motor mechanism, a fine-pitch ball screw mechanism, or the like, and delicate adjustment can be easily made.

After the collimation adjustment is thus finished, as illustrated in FIG. **8**A, while the collimator lens unit **30** is pressed in the arrow F direction by the adjust pin **5** to be pressed against the pair of bearing portions **13***c*, the adhesive **7** is filled in the open-hole **13***d* formed in the middle portion of the pair of bearing portions **13***c* in the module casing **10** and is cured.

If, for example, an ultraviolet curing adhesive is used as this adhesive **7**, it can be cured in a short time by being irradiated with ultraviolet light after being filled.

Owing to the curing of the adhesive **7**, the collimator lens unit **30** is bonded and fixed to the module casing **10** while being kept at such a predetermined position that part of its outer peripheral surface abuts on the pair of bearing portions **13***c,* and even if the pressing of the collimator lens unit **30** is released as illustrated in FIG. **8**B by pulling out the adjust pin **5**, the optical axis of the collimator lens **31** does not deviate or tilt.

Thereafter, when the adhesive is injected also from the long hole **13***b* for adjust pin insertion and is cured, it is possible to more surely fix the collimator lens unit **30** to the module casing **10**.

On the other hand, in a reference example, as illustrated in FIG. **11**A, it is assumed that an inner peripheral surface of a lens-barrel hole **10***a* of a module easing **10'** is formed as a round cylindrical surface without having bearing portions, and the adhesive **7** is filled and cured in a gap (clearance) between the inner peripheral surface and the outer peripheral surface of the collimator lens unit **30**.

In this case, as illustrated in FIG. **11**B, the collimator lens unit **30** is likely to be bonded and fixed to the lens-barrel hole **10***a* of the module casing **10'** in a state where the axis of the collimator lens unit **30**, that is, the optical axis of the built-in collimator lens is deviated from or tilted relatively to a central axis of the lens-barrel hole **10***a.*

This is because, due to the adhesive **7** filled in the clearance between the outer peripheral surface of the collimator lens unit **30** and the inner peripheral surface of the lens-barrel hole **10***a,* the collimator lens unit **30** is in a floating state in the lens-barrel hole **10***a,* and when the adhesive **7** is cured, contraction stresses are generated as illustrated by many arrows in FIG. **11**A, and due to unevenness or the like of an application amount of the adhesive, its strength differs depending on each circumferential-direction position.

Incidentally, in this embodiment, the collimator lens **31** and the aperture limit stop formation member **32** are integrally fixed to the cylindrical member **33** to form the collimator lens unit **30**, but instead, an aperture limit stop formation part and a cylindrical part may be integrally provided on the collimator lens itself to form the collimator lens unit. Further, the collimator lens and the cylindrical part may be integrally formed of the same material or different kinds of materials, and the aperture limit stop formation part may be integrally fixed to the resultant to form the collimator lens unit.

Incidentally, in the module **1** for the optical information reader of this embodiment, since the module casing **10** is formed of resin, the support shaft **44** of the vibration mirror **41** is liable to lack support strength if being supported in a cantilever manner. Here, a structure for enhancing the support strength will be described based on FIG. **3** and FIG. **9**. FIG. **9** is a sectional view taken along IX-IX line in FIG. **3**, and the vibration mirror driver **40** is entirely hatched in the same manner.

As illustrated in FIG. **9**, on the bottom surface portion **11** of the module casing **10** (including the vibration mirror driver attachment part **16**), the boss **11***a* is formed and a lower end portion of the support shaft **44** of the vibration mirror **41** is fit therein to be supported. An upper end portion of this support shaft **44** loosely penetrates through a through hole **70***a* formed in the circuit board **70** to protrude upward, and a holder disk **48** having a center hole is fit therearound as illustrated in FIG. **3** and FIG. **9** and this holder disk **48** is bonded or soldered to the upper surface of the circuit board **70** to be fixed.

With this structure, the support shaft **44** of the vibration mirror **41** is supported at two points by the bottom surface portion **11** of the module casing **10** and the circuit board **70**, which eliminates a risk of its tilting. Further, even when the support shaft **44** receives an external force such as a drop impact, a load for the bottom surface portion **11** of the module casing **10** to support the support shaft **44** is reduced.

Instead of the holder disk **48** having the center hole, a holder member in a hat shape having a recessed portion where to fit the upper end portion of the support shaft **44** and a flange portion may be used.

Next, FIG. **10** is a perspective view illustrating an embodiment in which a metallic foil is affixed on an outer wall surface of the module casing **10** to enhance a shielding effect, with the circuit board **70** removed.

In this embodiment using the photodiode (PD) as the light-receiving sensor **60**, in order to more ensure a noise

countermeasure, a metallic foil **9** is affixed on the outer wall surface, of the above-mentioned resin module casing **10**, at least near a portion housing the light-receiving sensor **60** having the built-in photodiode as illustrated in FIG. **10**, to thereby enhance the shielding effect.

Optical information read by the optical information reader including the module for the optical information reader according to the invention is not limited to bar codes but may be various kinds of two-dimensional codes such as PDF417, a QR code, and Aztec Code.

Hitherto, the embodiments of the invention have been described, but the invention is not limited to these, and it goes without saying that, in carrying out the invention, addition and changes can be appropriately made to their structures, part of the structures may be omitted, or shapes and materials may be changed.

The structures of the above-described embodiments and modification examples can of course be carried out by being arbitrarily combined as long as they are not mutually inconsistent.

INDUSTRIAL APPLICABILITY

The module for the optical information reader according to the invention is applicable to various kinds of optical information readers such as a bar code reader.

What is claimed is:

1. A module for an optical information reader in which a light-emitting unit having a light source, a collimator lens, a vibration mirror for scanning, a collector mirror or a condenser lens, and a light-receiving sensor are disposed in a module casing to be modularized,

wherein a collimator lens unit, in which an aperture limit stop formation member and the collimator lens are integrally disposed in a cylindrical member, is inserted in a lens-barrel hole of the module casing so as to be reciprocatable in an optical axis direction within a predetermined range, and the light-emitting unit is fixed to the module casing in the lens-barrel hole, with an optical axis of the light source aligned with an optical axis of the collimator lens, and wherein the aperture limit stop formation member radiates an output laser beam,

wherein a long hole through which an adjust pin is penetrated so as to be reciprocatable in the optical axis direction within a predetermined range is formed in a peripheral sidewall of the lens-barrel hole of the module casing, and a fitting portion in which a tip portion of the adjust pin penetrating through the long hole is fit is formed in an outer peripheral surface of the cylindrical member,

wherein, on an inner peripheral surface of the lens-barrel hole of the module casing, at a position diametrically opposite from the long hole and the fitting portion, a pair of bearing portions which, in a circumferential direction, are in point contact with the outer peripheral surface of the cylindrical member, and in an axial direction, are in line contact with the outer peripheral surface are formed at positions symmetrical with respect to the position in terms of an inner circumferential direction of the lens-barrel hole,

wherein an open-hole through which an adhesive for fixing the cylindrical member is fillable is formed in a middle region of the pair of bearing portions, and

wherein the bearing portions form a V-shaped slope by two flat surfaces which, in a circumferential direction, are in point contact with the outer peripheral surface of the cylindrical member, and in an axial direction, are in line contact with the outer peripheral surface.

2. The module for the optical information reader according to claim **1**, wherein an open-hole through which an adhesive for fixing the cylindrical member is fillable is formed in a middle region, of the V-shaped slope, which is not in contact with the outer peripheral surface of the cylindrical member.

3. The module for the optical information reader according to claim **1**, wherein the collimator lens unit is structured such that the collimator lens and the aperture limit stop formation member are fixed to the cylindrical member.

4. The module for the optical information reader according to claim **1**, wherein the collimator lens unit is structured such that the aperture limit stop formation member and the cylindrical member are integrally disposed on the collimator lens itself.

5. The module for the optical information reader according to claim **1**, wherein the collimator lens unit is structured such that the collimator lens and the cylindrical member are integrally formed of the same material or different kinds of materials, and to the resultant formed body, the aperture limit stop formation member is fixed.

6. The module for the optical information reader according to claim **1**, wherein the module casing is formed of resin.

7. The module for the optical information reader according to claim **6**, wherein the resin is reinforced resin in which carbon is dispersed.

8. The module for the optical information reader according to claim **6**, wherein a metallic foil is affixed on an outer wall surface of the module casing made of the resin, at least near a portion where the light-receiving sensor is housed.

* * * * *